UNIVERSIDAD
DE GRANADA

TESIS DOCTORAL

NEW APPLICATIONS OF MODELS BASED ON IMPRECISE PROBABILITIES WITHIN DATA MINING

Presented by:
Serafín Moral García

Advisors:
Joaquín Abellán Mulero
Carlos Javier Mantas Ruiz

To apply for the:
International PhD Degree in Information and Communication Technologies.

Programa de Doctorado en Tecnologías de la Información y la Comunicación

Noviembre 2022

Editor: Universidad de Granada. Tesis Doctorales
Autor: Serafín Moral García
ISBN: 978-84-1117-627-9
URI: https://hdl.handle.net/10481/79149

Serafín Moral García
New applications of models based on imprecise probabilities within Data Mining
Tesis Doctoral. ©© 2022

This document was written with LaTeX using a modified ArsClassica, a reworking of the ClassicThesis style designed by André Miede.

License

Este obra está bajo una licencia de **Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional**. https://creativecommons.org/licenses/by-nc-nd/4.0/

Contact

☞ seramoral@decsai.ugr.es

ABSTRACT

When we have information about a finite set of possible alternatives provided by an expert or dataset, a mathematical model is needed to represent such information. In some cases, a unique probability distribution is not appropriate for this purpose because the available information is not sufficient. For this reason, several mathematical theories and models based on imprecise probabilities have been developed in the literature. In this thesis work, we analyze the relations between some imprecise probability theories and study the properties of some models based on imprecise probabilities. When imprecise probability theories and models arise, tools for quantifying the uncertainty-based information in such theories and models, usually called uncertainty measures, are needed. In this thesis work, we analyze the properties of some existing uncertainty measures in theories based on imprecise probabilities and propose uncertainty measures in imprecise probability theories and models that present some advantages over the existing ones.

Situations in which it is necessary to represent the information provided by a dataset about a finite set of possible alternatives arise in classification, an essential task within Data Mining. This well-known task consists of predicting, for a given instance described via a set of attributes, the value of a variable under study, known as the class variable. In classification, it is often needed to quantify the uncertainty-based information about the class variable. For this purpose, classical probability theory (PT) has been employed for many years. In the last years, classification algorithms that represent the information about the class variable via imprecise probability models have been developed. Via experimental studies, it has been shown that classification methods based on imprecise probabilities significantly outperform the ones that utilize PT when data contain errors.

When classifying an instance, classifiers tend to predict a single value of the class variable. Nonetheless, in some cases, there is not enough information available for a classifier to point out a single class value. In these situations, it is more logical that classifiers predict a set of class values instead of a single value of the class variable. This is known as Imprecise Classification.

Classification algorithms (including Imprecise Classification) often aim to minimize the number of instances erroneously classified. This would be optimal if all classification errors had the same importance. Nevertheless, in prac-

tical applications, different classification errors usually lead to different costs. For this reason, classifiers that take the misclassification costs into account, also known as cost-sensitive classifiers, have been developed in the literature.

Traditional classification (including Imprecise Classification) assumes that each instance has a single value of a class variable. However, in some domains, this task does not fit well because an instance may belong to multiple labels simultaneously. In these domains, the Multi-Label Classification task (MLC) is more suitable than traditional classification. MLC aims to predict the set of labels associated with a given instance described via an attribute set. Most of the MLC methods proposed so far represent the information provided by an MLC dataset about the set of labels via classical PT.

In this thesis work, we develop new classification algorithms based on imprecise probability models, including Imprecise Classification, cost-sensitive Imprecise Classification, and MLC, that present some advantages and obtain better experimental results than the ones of the state-of-the-art.

RESUMEN AMPLIO EN CASTELLANO

Motivación

Cuando se dispone de información sobre un conjunto finito de posibles alternativas proveniente de un experto o un conjunto de datos, se suele emplear un modelo matemático para representar dicha información. Esto es muy útil para inferencia y toma de decisiones en el conjunto de alternativas. La teoría clásica de la probabilidad (TP) es la forma estándar de representar la información disponible sobre las alternativas. En la TP, la probabilidad de cada suceso puede determinarse de forma precisa. Este enfoque clásico es adecuado en muchos casos donde hay información suficiente para determinar la probabilidad de cada alternativa de forma exacta.

Sin embargo, en algunas situaciones, no hay información suficiente disponible para determinar la probabilidad de cada alternativa de manera precisa. Así, en tales casos, una única distribución de probabilidad no es apropiada para representar la información disponible en el conjunto de alternativas. Esto puede deberse a imprecisión o errores en la extracción de los datos. Por ejemplo, supongamos que tenemos una urna con diez bolas, cuatro de color negro, cuatro con color blanco y otras dos cuyo color no se sabe. En este caso, si extraemos aleatoriamente una bola de la urna, sabemos que la probabilidad de que esa bola sea negra es menor o igual que $0{,}6$ y mayor o igual que $0{,}4$, pero no podemos determinar de forma precisa la probabilidad de tal suceso, puesto que no sabemos cuántas de las otras dos bolas son negras.

Por este motivo, se han desarrollado muchas teorías matemáticas basadas en *probabilidades imprecisas* para representar la información disponible. Ejemplos de ellas son *conjuntos credales*, *probabilidades coherentes inferior y superior*, *capacidades de Choquet*, *teoría de la evidencia* (TE), e *intervalos de probabilidad alcanzables*. Todas estas teorías generalizan la TP. Algunas teorías de probabilidades imprecisas son más generales que otras, y hay también pares de teorías de probabilidades imprecisas tales que ninguna de ellas generaliza la otra. Una de las teorías más generales de probabilidad imprecisa es la basada en conjuntos credales [1]. Dado que cada teoría basada en probabilidades imprecisas tiene propiedades matemáticas específicas, algunas de estas teorías son más apropiadas que otras en situaciones concretas.

[1] Un conjunto credal es un conjunto cerrado y convexo de distribuciones de probabilidad.

Para representar la información sobre una muestra de observaciones acerca de un conjunto de alternativas, se han desarrollado en la literatura modelos matemáticos concretos basados en teorías de probabilidades imprecisas. Uno de los más remarcables es el *modelo impreciso de Dirichlet* (IDM). Este modelo es paramétrico y cumple algunos principios que se establecieron como apropiados para inferencia. Posteriormente, se propuso el *modelo no paramétrico de inferencia predictiva* (NPI-M). A diferencia del IDM, el NPI-M es un enfoque no paramétrico que no asume conocimiento previo sobre los datos. Además, las inferencias con el NPI-M suelen producir resultados intuitivamente más coherentes que las inferencias con el IDM. Pese a esto, el NPI-M necesita lidiar con restricciones difíciles debido a la representación de los datos usada en este modelo. De hecho, el conjunto de distribuciones de probabilidad compatibles con el NPI-M no es convexo. Para abordar este punto, se propuso el *modelo aproximado no paramétrico de inferencia predictiva* (A-NPI-M), que consiste en la envolvente convexa del conjunto de distribuciones de probabilidad consistentes con el NPI-M.

Cuando surgen teorías y modelos basados en probabilidades imprecisas, se hacen necesarias herramientas para cuantificar la información basada en incertidumbre en tales teorías modelos. Dichas herramientas se conocen como *medidas de incertidumbre*. La entropía de Shannon es la medida de incertidumbre bien establecida en la TP y es el punto de partida para medidas de incertidumbre en teorías más generales. En tales teorías, hay más tipos de incertidumbre que en la TP y, por consiguiente, es más difícil encontrar una medida de incertidumbre que satisfaga todas las propiedades esenciales. El estudio de medidas de incertidumbre en las teorías de probabilidades imprecisas más generales toma como referencia el estudio de medidas de incertidumbre en la TE. En esa teoría, la información puede representarse por una *asignación básica de probabilidad* (BPA) o, alternativamente, por una *función de creencia*. Hasta ahora, la entropía máxima en el conjunto de distribuciones de probabilidad compatibles con una función de creencia es la única medida de incertidumbre en la TE que satisface todas las propiedades y comportamientos requeridos. No obstante, el algoritmo propuesto hasta ahora para el cómputo de tal medida de incertidumbre es complejo. Por esta razón, se han propuesto muchas medidas alternativas en la TE, pero ninguna de estas medidas verifica las propiedades necesarias. Ha de señalarse que los intervalos de creencia para singletons, cuyas cotas inferior y superior son, respectivamente, los valores de probabilidad inferior y superior para singletons según la función de creencia, son más fáciles de manejar que las funciones de creencia para cuantificar la información basada en incertidumbre en la TE. De este modo, muchas de las alternativas a la entropía máxima propuestas durante los últimos años se basan en los intervalos de creencia para singletons. Sin embargo, cuando se usan los intervalos de creencia para singletons para representar la información basada en incertidumbre en vez de la función de creencia, se puede perder información. Con respecto a teorías generales basadas en probabilidades imprecisas, el máximo de entropía en conjuntos credales es una medida de incertidumbre bien establecida porque satisface las propiedades cruciales. Sin embargo, no hay algoritmo hasta ahora para calcular la entropía máxima en un conjunto credal general, aunque hay algoritmos para el cómputo de la entropía máxima en algunas teorías específicas de probabilidades imprecisas, como intervalos de probabilidad alcanzables y capacidades de Choquet de orden 2, una teoría bastante general. También se han propuesto algoritmos para la entropía máxima en modelos de probabilidad imprecisa como el IDM o el NPI-M.

Situaciones en las que se necesita representar la información sobre un conjunto de alternativas proporcionada por un conjunto de datos surgen en *clasificación*, un área esencial dentro de la *Minería de Datos*. Esta tarea consiste en predecir, para una instancia descrita mediante un conjunto de atributos, el valor de una variable bajo estudio llamada *variable clase*. Hoy en día, la clasificación se usa frecuentemente en muchas áreas. Los algoritmos de clasificación suelen representar la información dada por el conjunto de datos sobre la variable clase mediante un modelo matemático. Por lo tanto, en clasificación, a menudo se requiere cuantificar la información basada en incertidumbre sobre la variable clase.

Durante muchos años, la TP se ha empleado para representar la información sobre la variable clase en clasificación, considerando que la información dada por el conjunto de datos es suficiente para determinar la probabilidad de los valores clase de forma precisa. En los últimos años se han puesto algoritmos que representan la formación sobre la variable clase mediante modelos de probabilidades imprecisas. Tales algoritmos consideran que la información que hay en un conjunto de datos de clasificación es útil para aproximar la probabilidad de cada valor clase, pero no es suficiente para determinarla de forma precisa. Por medio de estudios experimentales, se ha mostrado que los métodos de clasificación basados en probabilidades imprecisas rinden significativamente mejor que los que utilizan la TP cuando hay ruido de la variable clase[2] en los datos.

Los clasificadores suelen intentar minimizar el número de predicciones incorrectas. Este punto es óptimo cuando todos los errores de clasificación tienen la misma importancia. No obstante, en aplicaciones prácticas, diferentes errores de clasificación a menudo implican costes diferentes. Por ejemplo, en

[2] En clasificación, el término 'ruido' se usa para referirse a errores en los datos.

diagnóstico médico, las consecuencias de predecir incorrectamente que un paciente no tiene una enfermedad seria son probablemente mucho peores que las consecuencias de predecir erróneamente que el paciente no tiene dicha enfermedad; en *predicción de software defectuoso*, el coste de módulos defectuosos predichos como no defectuosos podría ser más alto que el coste de módulos no defectuosos predichos como defectuosos; en *detección de fraudes de crédito*, predecir que una tarjeta de crédito fraudulenta es legal probablemente cause pérdidas económicas mucho mayores para bancos e instituciones financieras que predecir una tarjeta de crédito normal como fraudulenta. Por consiguiente, se han desarrollado clasificadores que tienen en cuenta los costes de errores, conocidos como *clasificadores sensibles al coste*.

Cuando se clasifica una instancia, los clasificadores sensibles e insensibles al coste normalmente predicen un único valor de la variable clase. Sin embargo, en algunas situaciones, la información dada por el conjunto de datos no es suficiente para que un clasificador señale a un único valor clase. En estos casos, es probablemente más lógico que los clasificadores predigan un conjunto de valores de la variable clase, lo que se conoce como *clasificación imprecisa*. Por ejemplo, supongamos que, en un conjunto de datos de clasificación para medicina, hay cinco valores clase, correspondiente a cinco enfermedades que puede tener un paciente. Es posible que, para predecir la enfermedad de un paciente, la información dada por el conjunto de datos solo nos permita saber que el paciente puede tener tres de las cinco enfermedades, pero no hay información suficiente para determinar cuál de esas tres enfermedades tiene el paciente. En esta situación, aunque la enfermedad de paciente no puede determinarse de forma precisa, la información dada puede ser útil para conocer un tratamiento adecuado para el paciente. No obstante, podría ser arriesgado predecir una de esas tres enfermedades porque, en caso de error, el tratamiento podría tener consecuencias negativas. Intuitivamente, una métrica de evaluación para un algoritmo de clasificación imprecisa ha de considerar si las predicciones son correctas (el valor clase real está entre los predichos), y cómo de informativas son las predicciones, lo que se mide por el número medio de valores clase predichos.

Para desarrollar métodos de clasificación imprecisa, las teorías basadas en probabilidades imprecisas son más apropiadas que la TP. Hasta ahora se han propuesto pocos algoritmos de clasificación imprecisa. El primero fue el Naïve Credal Classifier (NCC), que combina el IDM con la suposición naïve (todos los atributos son independientes dada la variable clase) para dar predicciones imprecisas. Posteriormente, se propuso un algoritmo de clasificación imprecisa basado en árboles de decisión, llamado el Imprecise Credal Decision Tree

(ICDT). Tanto el NCC como el ICDT se adaptaron para clasificación sensible al coste.

La clasificación tradicional (incluyendo clasificación imprecisa) supone que cada instancia tiene único valor de una variable clase. No obstante, en algunos dominios, esta tarea no encaja bien porque una instancia puede pertenecer a múltiples etiquetas simultáneamente. Por ejemplo, en *categorización de texto*, si un texto trata sobre la visita de Donald Trump a Francia, tiene sentido que tal texto pertenezca a las etiquetas 'Estados Unidos' y'Francia'; en *biología*, una proteína puede tener múltiples funciones en el cuerpo humano; en una *imagen* o fragmento de *música* pueden aparecer varias emociones. En estos dominios, la tarea de *Clasificación multi-etiqueta* (MLC) es más adecuada que la clasificación tradicional. La MLC trata de predecir el conjunto de etiquetas asociadas a una instancia dada descrita mediante un conjunto de atributos.

Se han desarrollado hasta ahora muchos enfoques a MLC. Los métodos para MLC pueden dividirse en dos grupos. Por un lado, los *métodos de transformación del problema* convierten la tarea de MLC en múltiples problemas de clasificación tradicional y combinan sus soluciones para dar una salida a la tarea de MLC. Por otro lado, los *métodos de adaptación de algoritmo* adaptan directamente los algoritmos existentes para clasificación tradicional a MLC. Muchos de estos métodos representan la información dada por un conjunto de datos de MLC sobre el conjunto de etiquetas mediante la TP. Dado que el número de etiquetas en MLC suele ser muy grande, explotar correlaciones entre etiquetas es un reto importante para los algoritmos de MLC. Hay algunos enfoques para determinar correlaciones entre etiquetas en MLC basados en probabilidades precisas. Las cadenas de clasificadores son considerados métodos simples y efectivos para explotar correlaciones entre etiquetas en MLC. Estos métodos, para cada etiqueta, tienen en cuenta las predicciones realizadas para las etiquetas anteriores según un orden establecido. Dicho orden influye mucho en el rendimiento de una cadena de clasificadores. Además, en MLC, a menudo muy pocas instancias pertenecen a una cierta etiqueta. En consecuencia, los algoritmos para MLC suelen sufrir un problema de no balanceo de clases.

Objetivos

En esta tesis seguimos la línea de investigación de teorías y modelos de probabilidades imprecisas y medidas de incertidumbre con probabilidades imprecisas. También proponemos nuevos métodos de clasificación basados en probabilidades imprecisas que obtienen mejor rendimiento que los del estado del arte.

Hay cinco objetivos principales de esta tesis doctoral, los cuales pueden dividirse en objetivos específicos:

1. En primer lugar, tratamos de analizar las propiedades y relaciones entre algunas teorías y modelos de probabilidades imprecisas. Este objetivo se divide en dos:

   a) Caracterizar los conjuntos credales representables mediante funciones de creencia e intervalos de probabilidad alcanzables, dos teorías de probabilidades imprecisas tales que ninguna de ellas generaliza la otra. Para esto, nuestro objetivo es dar un conjunto de condiciones necesarias y suficientes bajo las cuales un conjunto de intervalos de probabilidad alcanzables es representable por una función de creencia, así como una caracterización de funciones de creencia representables por intervalos de probabilidad alcanzables.

   b) Analizar las propiedades principales de conjuntos credales asociados al A-NPI-M, comparándolas con las propiedades de conjuntos credales correspondientes al IDM.

2. Con respecto a medidas de incertidumbre, nuestro objetivo principal es analizar las propiedades de algunas medidas de incertidumbre en teorías y modelos de probabilidades imprecisas y proponer medidas de incertidumbre en probabilidades imprecisas que presenten algunas ventajas sobre las existentes. Específicamente, hay cuatro objetivos relacionados con medidas de incertidumbre:

   a) Hacer un análisis crítico de alternativas recientes a la entropía máxima en la TE mediante sus propiedades y comportamientos.

   b) Estudiar las propiedades matemáticas y requisitos de comportamiento esenciales para medidas de incertidumbre en intervalos de creencia para singletons. También tratamos de analizar cuáles de esas propiedades matemáticas y requisitos de comportamiento satisfacen cada una de las medidas de incertidumbre en intervalos de creencia para singletons propuestas hasta ahora.

   c) Presentar una medida de incertidumbre en intervalos de creencia para singletons que, a diferencia de las propuestas hasta ahora, satisfaga todas las propiedades matemáticas y requisitos de comportamiento fundamentales para este tipo de medida. Además, pretendemos que, en aplicaciones prácticas, la medida propuesta sea más fácil de manejar que la entropía máxima en una BPA, la medida de incertidumbre bien establecida en la TE.

   d) Proponer procedimientos para el cómputo de las principales medidas de incertidumbre en conjuntos credales derivados del A-NPI-M.

3. Tratamos de desarrollar un nuevo método de clasificación tradicional basado en modelos de probabilidades imprecisas que logre mejores resultados que la versión existente de tal algoritmo basada en la TP, especialmente cuando hay ruido de la variable clase en los datos.

4. En cuanto a la clasificación imprecisa, tratamos de desarrollar mejoras sobre los algoritmos propuestos hasta ahora en este campo. Concretamente, los objetivos vinculados a clasificación imprecisa pueden resumirse como sigue:

   a) Presentar un nuevo Imprecise Credal Decision Tree que use el A-NPI-M, a diferencia del ya existente, que emplea el IDM. Nuestra idea es mostrar que el A-NPI-M obtiene resultados estadísticamente equivalentes al IDM con la mejor elección del parámetro cuando se emplean ambos modelos en el Imprecise Credal Decision Tree.

   b) Proponer una nueva versión del algoritmo NCC que lleve a predicciones mucho más informativas que el NCC existente.

   c) Desarrollar el primer método ensemble para clasificación imprecisa. Hemos de remarcar que, como los clasificadores imprecisos suelen proporcionar como salida un conjunto de valores clase, no es trivial combinar múltiples predicciones imprecisas. Esta puede ser la razón por la que no se ha propuesto hasta ahora ningún método ensemble para clasificación imprecisa. Por lo tanto, para desarrollar un ensemble de clasificadores imprecisos tenemos que proponer una técnica para combinar múltiples predicciones imprecisas.

   d) Con respecto a clasificación imprecisa sensible al coste, nuestro objetivo es proponer un nuevo Imprecise Credal Decision Tree sensible al coste que presente algunas ventajas y logre mejores resultados que el ya existente.

5. Nuestro último objetivo es presentar nuevos métodos para MLC basados en modelos de probabilidades imprecisas que rindan mejor que los existentes basados en probabilidades precisas, siendo la mejora más notable conforme hay más ruido en las etiquetas. Este objetivo se divide en los cuatro siguientes:

a) Analizar el uso de probabilidades imprecisas en dos métodos para MLC de transformación del problema, señalando que supone una mejora sobre la TP, especialmente con ruido en las etiquetas.

b) Proponer una nueva adaptación de árboles de decisión para MLC que use probabilidades imprecisas, a diferencia de la ya existente, basada en la TP. Nuestro objetivo es mostrar que nuestra adaptación propuesta es menos sensible al ruido en las etiquetas que la propuesta hasta ahora.

c) Presentar nuevos algoritmos lazy para MLC que empleen probabilidades imprecisas, a diferencia de algunos métodos lazy para MLC desarrollados hasta ahora, que usan la TP. Tratamos de demostrar teórica y empíricamente que nuestros métodos lazy para MLC propuestos son más adecuados que los métodos lazy para MLC existentes basados en la TP para abordar en problema de no balanceo de clases que aparece frecuentemente en MLC, especialmente con ruido en las etiquetas.

d) Proponer un nuevo método para explotar correlaciones entre etiquetas en MLC basado en modelos de probabilidades imprecisas. Tratamos de ilustrar que nuestro método propuesto presenta algunas ventajas sobre otros algoritmos existentes para explotar correlaciones entre etiquetas en MLC basados en la TP. La idea es corroborar este punto mediante un análisis experimental.

Finalmente, también tratamos de aplicar modelos de probabilidades imprecisas a dominios importantes como *análisis de riesgo de crédito* y *análisis de accidentes de tráfico* para extraer conocimiento útil en tales dominios.

Estructura de la tesis

Esta tesis doctoral se divide en cuatro partes más un apéndice. Cada parte se subdivide en capítulos.

- En la primera parte contextualizamos nuestro trabajo y establecemos nuestros objetivos principales (capítulo 1).

- La parte ii describe el conocimiento previo necesario para nuestro trabajo de tesis. Esta parte se divide en cinco capítulos. En el capítulo 2 se describen las principales teorías y modelos de probabilidades imprecisas usados en este trabajo. El capítulo 3 da una visión general de las principales medidas de incertidumbre en probabilidades imprecisas propuestas hasta ahora. En el capítulo 4 exponemos la tarea de clasificación, así como los enfoques a esta tarea considerados en este trabajo de tesis. La tarea de clasificación imprecisa y los métodos propuestos hasta ahora para tal tarea se detallan en el capítulo 5. El capítulo 6 describe la tarea de clasificación multi-etiqueta y los enfoques principales a este ámbito.

- Las contribuciones de esta tesis se presentan en la parte iii. Dicha parte se divide en cinco capítulos. Algunas teorías y modelos de probabilidades imprecisas se analizan en el capítulo 7, el cual se corresponde con el primer objetivo. En el capítulo 8, que se asocia con el segundo objetivo, analizamos algunas medidas de incertidumbre en teorías y modelos de probabilidades imprecisas y proponemos medidas de incertidumbre en tales teorías y modelos. En el capítulo 9 se presenta un nuevo método de clasificación tradicional basado en probabilidades imprecisas. Tal capítulo se asocia con el tercer objetivo. El capítulo 10, correspondiente al cuarto objetivo, detalla nuestros algoritmos propuestos para clasificación imprecisa. Nuestros métodos propuestos para clasificación multi-etiqueta basados en modelos de probabilidades imprecisas se presentan en el capítulo 11, el cual se asocia con el quinto objetivo.

- Las conclusiones e ideas para trabajo futuro se dan en la parte iv (capítulo 12).

Finalmente, en el apéndice A, mostramos la aplicación de algunos modelos de probabilidad imprecisa para extraer conocimiento útil en algunos dominios importantes como análisis de riesgo de crédito y análisis de accidentes de tráfico.

Conclusiones

A continuación describimos las contribuciones principales de este trabajo de tesis.

- Se han caracterizado los conjuntos credales representables por funciones de creencia e intervalos de probabilidad alcanzables: hemos dado un conjunto de condiciones necesarias y suficientes que ha de cumplir un conjunto de intervalos de probabilidad alcanzables para ser representable mediante una función de creencia. Se ha demostrado que, para comprobar dichas condiciones, se requiere considerar varios subconjuntos y comprobar algunas desigualdades simples con las sumas de las probabilidades inferiores y superiores en dichos subconjuntos. Los conjuntos se obtienen también de forma fácil y rápida. También hemos dado una caracterización de funciones de creencia representables por medio de intervalos de probabilidad alcanzables. En concreto, se ha demostrado que la condición necesaria y suficiente para que una función de creencia sea representable por un conjunto de intervalos de probabilidad alcanzables es la siguiente: la diferencia entre cualquier par de elementos focales de la correspondiente asignación básica de probabilidad de cardinalidad mayor o igual que 2 tiene una cardinalidad menor o igual a uno. Usando nuestra condición dada, hemos caracterizado algunos tipos especiales de funciones de creencia, como p-boxes o medidas de necesidad, que pueden representable mediante conjuntos de intervalos de probabilidad alcanzables.

- Con respecto a modelos de probabilidades imprecisas, hemos analizado las propiedades principales de los conjuntos credales derivados del A-NPI-M, comparándolas con conjuntos credales asociados al IDM. Se ha mostrado que, al igual que con el IDM, a medida que el tamaño muestral converge a infinito, los conjuntos credales relacionados con el A-NPI-M convergen a una única distribución de probabilidad, que se obtiene mediante frecuencias relativas; el A-NPI-M es un modelo más impreciso que el IDM con el valor más utilizado del parámetro, el recomendado en la literatura; una de las propiedades más remarcables de los conjuntos credales correspondientes al A-NPI-M es que no siempre pueden representarse por una función de creencia, a diferencia de los conjuntos credales derivados del IDM. El cálculo de la inversa de Möbius para el A-NPI-M es más complejo que para el IDM. Lo mismo ocurre con el conjunto de puntos extremos del conjunto credal. Por lo tanto, el A-NPI-M es un modelo más complejo que el IDM. No obstante, ha de remarcar que el IDM supone conocimiento previo de los datos mediante un parámetro, a diferencia del A-NPI-M.

- En cuanto a medidas de incertidumbre, hemos hecho un análisis crítico de dos alternativas a la entropía máxima en la TE propuestas hace unos años. Se ha probado que dichas medidas no satisfacen muchas de las propiedades matemáticas cruciales para medidas de incertidumbre en la TE, y su comportamiento en algunos escenarios es también cuestionable. Además, hemos realizado un estudio sobre las propiedades matemáticas y requisitos de comportamiento esenciales para medidas de incertidumbre en intervalos de creencia para singletons. Dicho estudio se ha basado en el llevado a cabo previamente para medidas de incertidumbre en BPAs. Hemos mostrado que ninguna de las medidas de incertidumbre en intervalos de creencia para singletons propuestas hasta ahora verifica todas las propiedades matemáticas y requisitos de comportamiento fundamentales para este tipo de medida. También hemos propuesto una medida de incertidumbre en intervalos de creencia para singletons que consiste en la entropía máxima en el conjunto credal asociado a dichos intervalos. Hemos demostrado que, pese a que nuestra medida propuesta requiere un cómputo más complejo que las otras medidas de incertidumbre en intervalos de creencia para singletons propuestas hasta ahora, es la única que satisface todas las propiedades matemáticas y requisitos de comportamiento cruciales para medidas de incertidumbre en intervalos de creencia para singletons. También hemos señalado que nuestra medida propuesta da una cota superior de la entropía máxima en el conjunto credal compatible con una BPA, la medida de incertidumbre bien establecida en la TE, siendo el cómputo de la primera medida notablemente más rápido que el de la segunda. Además, hemos mostrado cómo calcular las medidas de incertidumbre más importantes en conjuntos credales asociados con el A-NPI-M. Esto hace que el A-NPI-M sea muy útil para aplicaciones prácticas.

- Dentro de la clasificación tradicional, hemos presentado una nueva versión del algoritmo Naïve Bayes (NB), llamado el Imprecise m-probability estimation Naïve Bayes (ImNB), que considera las probabilidades a prior de los valores clase para estimar las probabilidades condicionales, como una versión del NB propuesta hace unos años. Sin embargo, ImNB usa la medida de incertidumbre bien establecida en conjuntos credales para estimar las probabilidades a priori, a diferencia de modelos previos, que usan frecuencias relativas con corrección de Laplace para estimar tales probabilidades. Por consiguiente, nuestro ImNB propuesto es más robusto al ruido en la variable clase que las versiones del algoritmo NB propuestas previamente. Un estudio experimental con varios niveles de ruido ha puesto de manifiesto que ImNB obtiene mejor rendimiento que las versiones del algoritmo NB que usan estimaciones clásicas de las probabilidades, con y sin ruido en los datos.

- Se han propuesto mejoras sobre los métodos de clasificación imprecisa desarrollados hasta ahora. Específicamente, podemos resumir las contribuciones de esta tesis vinculadas a clasificación imprecisa en los siguientes puntos:

- Hemos propuesto una nueva versión del algoritmo ICDT que emplea el A-NPI-M para el criterio de ramificación y para los intervalos de probabilidad en los nodos hoja (ICDT-ANPI), mientras que el ICDT existente usa el IDM. Resultados experimentales han mostrado que el ICDT-ANPI obtiene un rendimiento equivalente al ICDT con la mejor elección del parámetro para el IDM. En consecuencia, el A-NPI-M es más apropiado que el IDM para árboles de decisión para clasificación imprecisa porque el primer modelo no supone conocimiento previo sobre los datos mediante un parámetro, a diferencia del segundo.

- Se ha desarrollado una nueva versión del algoritmo NCC llamada Extreme Prior Naïve Credal Classifier (EP-NCC). A diferencia de NCC, EP-NCC tiene en cuenta las probabilidades a prior inferior y superior de los valores clase para estimar las probabilidades condicionales inferior y superior. Se ha mostrado que las predicciones hechas por EP-NCC son probablemente más informativas que las realizadas por NCC, no siendo el riesgo de predicciones incorrectas mucho más alto con EP-NCC. Un análisis experimental ha revelado que EP-NCC rinde significativamente mejor que NCC dado que el primer método es mucho más informativo que el segundo, mientras que la diferencia entre el rendimiento de ambos algoritmos es acierto no es estadísticamente significativa. El análisis experimental también ha puesto de manifiesto que EP-NCC e ICDT obtienen rendimiento equivalente, pero ICDT requiere un tiempo computacional mucho más alto que EP-NCC. Así, debido al buen rendimiento y al bajo tiempo computacional, EP-NCC es más adecuado para grandes conjuntos de datos de clasificación imprecisa que los algoritmos existentes para tal tarea. Este es un punto importante a favor de nuestro algoritmo propuesto EP-NCC a causa de la creciente cantidad de datos en cualquier área.

- En este trabajo de tesis se ha presentado el primer método ensemble para clasificación imprecisa. Se ha tenido en cuenta que el esquema Bagging ha obtenido buen rendimiento en clasificación precisa, especialmente cuando se usa con árboles de decisión credales (CDTs), lo que fomenta diversidad. Así, nuestro método ensemble propuesto para clasificación imprecisa consiste en un esquema Bagging que usa el algoritmo ICDT (la adaptación de CDT a clasificación imprecisa) como clasificador base (Bagging-ICDT). La clave es cómo combinar las predicciones realizadas por múltiples clasificadores imprecisos. Esto no es trivial, ya que, si las predicciones imprecisas no se combinan adecuadamente, el ensemble podría no rendir mejor que un clasificador individual porque puede producirse una reducción de información excesiva. Nuestra técnica de combinación propuesta intenta que el clasificador Bagging impreciso sea lo más informativo posible. Dicha técnica consiste en predecir como no dominados solo aquellos valores clase con el valor mínimo posible de dominancia, lo que implica que no es muy conservativa. Mediante un análisis experimental, hemos mostrado que el Bagging-ICDT con nuestra técnica de combinación propuesta obtiene mejor rendimiento que el método ICDT; Bagging-ICDT es más informativo que ICDT, mientras que la diferencia entre el rendimiento de ambos algoritmos en hacer predicciones correctas no es significativa.

  - Con respecto a la clasificación sensible al coste, hemos propuesto un nuevo Imprecise Credal Decision Tree sensible al coste que pondera las instancias considerando el coste de clasificar erróneamente el valor clase correspondiente. Nuestro método propuesto tiene en cuenta los costes de errores en el proceso de construcción del árbol, a diferencia del cost-sensitive Imprecise Credal Decision Tree existente, que solo considera los costes de errores al clasificar instancias en nodos hoja. Hemos mostrado que el criterio que usa nuestro Imprecise Credal Decision Tree sensible al coste propuesto para clasificar instancias en nodos hoja es probablemente más efectivo que el empleado por el Imprecise Credal Decision Tree sensible al coste existente porque las predicciones son posiblemente más informativas. Un estudio experimental ha puesto de manifiesto que nuestro Imprecise Credal Decision Tree sensible al coste propuesto rinde significativamente mejor que el ya existente; aunque el coste de clasificación incorrecta de nuestro método propuesto es más alto, nuestro algoritmo propuesto es mucho más informativo y logra un mejor compromiso entre bajo coste de predicciones erróneas y predicciones informativas. De este modo, concluimos que nuestro Imprecise Credal Decision Tree sensible al coste propuesto es más apropiado que el ya existente para aplicaciones prácticas donde los costes de errores son diferentes y la información disponible no es suficiente para que los clasificadores predigan un único valor clase.

- También hemos propuesto nuevos algoritmos para MLC basados en modelos de probabilidad imprecisa. Hemos mostrado que el ruido intrínseco de etiquetas en MLC es probablemente más alto que el ruido intrínseco de clase en clasificación tradicional. En consecuencia, puesto que los algoritmos que usan probabilidades imprecisas obtienen mejor rendimiento que los basados en la TP cuando hay ruido de clase en los datos, nuestros métodos propuestos para MLC son probablemente más adecuados que los desarrollados hasta ahora basados en probabilidades precisas. Hemos comprobado este punto mediante estudios experimentales. En concreto, las contribuciones de esta tesis con respecto a MLC pueden resumirse en los siguientes puntos:

– En primer lugar, hemos analizado el uso de CC4.5 en dos métodos de transformación del problema: Binary Relevance (BR) y Calibrated Label Ranking (CLR). BR es un método de MLC muy simple que ha obtenido buen rendimiento en la práctica, y CLR explota correlaciones entre etiquetas por pares y alivia el problema de no balanceo de clases que suele aparecer en MLC. Hemos mostrado que, como CC4.5 es más robusto al ruido de clase que C4.5, BR y CLR son menos sensibles al ruido en las etiquetas con CC4.5 que con C4.5. Así, puesto que el ruido intrínseco de etiquetas en MLC es probablemente mayor que el ruido intrínseco de clase en clasificación tradicional, CC4.5 es probablemente más adecuado que C4.5 para abordar los problemas de clasificación binaria en BR y CLR. Resultados experimentales han mostrado que tanto BR como CLR obtienen mejor rendimiento con CC4.5 que con C4.5, siendo la mejora más notable a medida que el ruido en las etiquetas es mayor.

– Hemos propuesto una nueva adaptación de árboles de decisión para MLC que emplea el A-NPI-M para el criterio de ramificación y para predecir las probabilidades sobre la relevancia de las etiquetas para las instancias en los nodos hoja. Hemos mostrado que nuestra adaptación propuesta es menos sensible al ruido en las etiquetas que la propuesta hasta ahora, la cual se basa en la TP. Resultados experimentales han señalado que nuestra adaptación propuesta obtiene mejor rendimiento que la existente, siendo la mejora más notable a medida que hay más ruido en las etiquetas. Por lo tanto, el A-NPI-M es más apropiado que la TP para usarse en las adaptaciones de árboles de decisión para MLC, especialmente cuando hay ruido en las etiquetas.

– También hemos presentado dos algoritmos lazy para MLC que, para clasificar una instancia, emplean estimadores estadísticos basados en las instancias vecinas, de forma similar a algunos algoritmos lazy existentes para MLC. Sin embargo, nuestros métodos lazy propuestos usan el A-NPI-M para dichos estimadores estadísticos, a diferencia de los existentes, que utilizan frecuencias relativas con corrección de Laplace. Hemos mostrado que nuestros algoritmos lazy propuestos para predicen que una etiqueta es relevante para una instancia más frecuentemente que los existentes, y este hecho se enfatiza con ruido en las etiquetas. Un estudio experimental ha revelado que nuestros métodos lazy propuestos para MLC son más apropiados que los existentes basados en probabilidades precisas para tratar el problema de no balanceo de clases que suele surgir en MLC, especialmente con ruido en las etiquetas.

- Finalmente, hemos propuesto un procedimiento de ordenación de etiquetas en cadenas de clasificadores que estima la correlación entre cada par de etiquetas mediante el A-NPI-M y ordena las etiquetas con un procedimiento greedy. En dicho procedimiento, para cada etiqueta candidata, se considera la correlación media entre esa etiqueta y las ya insertadas, así como la correlación media entre esa etiqueta y las no insertadas aún. Se ha mostrado que nuestro procedimiento propuesto presenta algunas ventajas sobre los desarrollados hasta ahora basados en correlaciones entre etiquetas; emplea un modelo de probabilidades imprecisas para estimar correlaciones entre etiquetas, que es más apropiado que probabilidades precisas; nuestro método propuesto, para cada etiqueta candidata, tiene en cuenta la correlación de ella con las etiquetas ya insertadas y la correlación de las etiquetas no insertadas con la etiqueta candidata, mientras que alguno de los métodos de ordenación propuestos hasta ahora solo consideran las correlaciones entre la etiqueta candidata y las no insertadas aún. Un estudio experimental ha mostrado que nuestro método propuesto de ordenación obtiene mejor rendimiento que los basados en correlaciones entre etiquetas desarrollados hasta ahora.

AGRADECIMIENTOS

El trabajo que aquí se presenta se ha financiado mediante una beca de Formación para el Profesorado Universitario (FPU) proporcionada por el Ministerio de Educación, Formación Profesional y Universidades. La referencia de dicha beca es FPU17/02685. El trabajo de esta tesis también ha sido financiado por el Ministerio Español de Economía y Competitividad y por el "Fondo Europeo de Desarrollo Regional"(FEDER) con el proyecto TEC2015-69496-R, por UGR-FEDER con el proyecto A-TIC-344-UGR20 y por la Junta de Andalucía-Consejería de Transformación Económica, Industria, Conocimiento y Universidades con el proyecto P20_00159.

Largo ha sido el trabajo llevado a cabo durante años para esta tesis doctoral. Muchas son las personas a las que debo de agradecer su apoyo sin el cual no habría sido posible terminar este trabajo.

En primer lugar, quiero agradecer a mis directores Carlos Javier Mantas y Joaquín Abellán por haberme dado la oportunidad de trabajar con vosotros desde el año 2016, por todas las herramientas que me habéis facilitado y por todo el tiempo dedicado a guiarme y a revisar mi trabajo. También quiero agradecer a Francisco Javier García Castellano, que también ha colaborado muy activamente en trabajos relacionados con el equipo de investigación de la tesis y, durante los años en los que yo he trabajado con dicho grupo, ha dedicado también bastante tiempo en la revisión de mis trabajos. Todos estos años trabajando con este equipo me han aportado una gran formación laboral, intelectual, y también como persona. Y espero que esto no se quede aquí.

No me puedo olvidar tampoco de agradecer a la Universidad de Granada por toda la formación que he ha dado desde que entré como alumno en el año 2011. En concreto, también quiero agradecer al Departamento de Ciencias de la Computación e Inteligencia Artificial, donde he encontrado profesores de los que he aprendido bastante y donde he tenido la suerte de poder tener mis primeras experiencias docentes, un trabajo que me ha encantado y con el que espero poder seguir en los próximos años.

Quiero agradecer también a los que me han dirigido las dos estancias que he realizado durante el período de tesis doctoral por lo bien que me han guiado y su tiempo y esfuerzo dedicado a la revisión de mi trabajo durante dichas estancias. Me refiero a Sebastien Destercke, de la Universidad de Compiegne (Francia) y a Frank P. Coolen, de la Universidad de Durham (Reino Unido).

Como no podría ser de otro modo, quiero agradecer también a mis padres, que me han apoyado desde el primer día hasta el último de mi vida. Sin ellos no habría sido posible terminar este trabajo, por no decir ni empezarlo.

Finalmente, quiero agradecer al resto de familiares y a mis amigos, por preocuparos por mí constantemente y apoyarme en los momentos más difíciles de mi vida.

Muchísimas gracias

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION

1 INTRODUCTION AND OBJECTIVES

1.1 Overview

When there is information about a finite set of possible alternatives provided by an expert or dataset, a mathematical model is often employed to represent such information. This is very useful for inferences and decision-making in the set of alternatives. *Classical probability theory* (PT) is the standard way of representing the information involved in the set of alternatives. In PT, the probability of each event can be precisely determined. This classical approach is suitable in many situations where there is sufficient information to determine the probability of each alternative in an exact way.

Nevertheless, in some cases, there is not enough information available to precisely determine the probability of each alternative. Hence, in such situations, a single probability distribution is not sufficient for representing the information available in the set of alternatives. This can be due to imprecision or errors in the extraction of the data. For example, suppose that we have an urn with ten balls, four with the color white, four with the color black, and the other two with unknown color. In this case, if we randomly extract a ball from the urn, we know that the probability of such a ball being black is lower or equal to 0.6 and greater or equal to 0.4, but we cannot precisely determine the probability of such an event since we do not know how many of the other two balls are black.

For this reason, many mathematical theories based on *imprecise probabilities* have been developed in the literature to represent the available information. Examples are *credal sets*, *coherent lower* and *upper probabilities*, *Choquet capacities*, *evidence theory* (ET), and *reachable probability intervals*. All these theories generalize PT. Some imprecise probability theories are more general than others, and there are also pairs of imprecise probability theories such that any of them generalizes the other. One of the most general imprecise probability theories is the one based on credal sets[1]. As each imprecise probability theory has specific mathematical properties, some theories are more appropriate than others in specific situations. A detailed description of these theories can be found in [208].

[1] A credal set is a closed and convex set of probability distributions.

In order to represent the information involved in a sample of observations about the set of alternatives, specific mathematical models based on imprecise probability theories have been developed in the literature. One of the most remarkable is the parametric *Imprecise Dirichlet Model* (IDM) [209]. This model satisfies some principles that were claimed to be desirable for inference. Afterwards, the *Non-Parametric Predictive Inference Model* (NPI-M) was proposed [58, 59]. Unlike the IDM, the NPI-M is a non-parametric approach that does not assume previous knowledge about the data. Moreover, inferences made with the NPI-M tend to produce intuitively more coherent results than inferences made with the IDM [59]. Even so, the NPI-M needs to deal with difficult constraints due to the representation of the data used in this model. In fact, the set of probability distributions compatible with the NPI-M is not convex. In order to handle this issue, the *Approximate Non-Parametric Predictive Inference Model* (A-NPI-M) was developed in [5], which consists of the convex hull of the set of probability distributions consistent with the NPI-M.

When imprecise probability theories and models arise, tools for quantifying the uncertainty-based information in such theories and models are needed. Such tools are known as *uncertainty measures*. The Shannon entropy [189] is the well-established uncertainty measure in PT and is the starting point for uncertainty measures in more general theories than PT. In such theories, there are more types of uncertainty than in PT and, thus, it is more difficult to find an uncertainty measure that satisfies all essential properties. The study of uncertainty measures in the most general imprecise probability theories takes the study of uncertainty measures in ET as a reference. In that theory, the information can be represented by a *basic probability assignment* (BPA) or, alternatively, by a *belief function*. So far, the maximum entropy on the set of probability distributions compatible with a belief function is the only uncertainty measure in ET that satisfies all required mathematical properties and behaviors [21]. Nonetheless, the algorithm proposed so far to compute such an uncertainty measure is complex. For this reason, many alternative measures have been proposed in ET, but none of these measures verifies the necessary properties. It must be remarked that the belief intervals for singletons, whose lower and upper bounds are, respectively, the lower and upper probability values for the singletons according to the belief function, are easier to manage than belief functions for quantifying uncertainty-based information in ET. In this way, many alternative measures to the maximum entropy proposed during the last years are based on belief intervals for singletons. However, when belief intervals for singletons are used to represent the uncertainty-based information instead of the belief function, some information may be lost. Concerning general imprecise probability theories, the maximum entropy on credal sets is a well-established uncertainty measure because it satisfies the crucial properties. Nonetheless, there is no algorithm so far to compute the maximum entropy on a general credal set, even though there are algorithms for computing the maximum entropy in some specific imprecise probability theories, such as reachable probability intervals [13] or Choquet capacities of order 2 [16], a quite general imprecise probability theory. Algorithms for the maximum entropy in imprecise probability models such as the IDM or the NPI-M have also been proposed [2, 5].

Situations in which it is necessary to represent the information about a finite set of possible alternatives provided by a dataset arise in *classification* [107], an essential area within *Data Mining*. This well-known task consists of predicting, for a given instance described via a set of attributes or features, the value of a variable under study called the *class variable*. Nowadays, classification is commonly used in many domains. For example, in *medicine*, this task is widely employed to predict whether a patient has a disease by using a set of features of such a patient; in *credit fraud detection*, classification is suitable for detecting whether a card is fraudulent by utilizing a set of attributes of the card. Classification algorithms usually represent the information provided by a dataset about the class variable by means of a mathematical model. Thereby, in classification, it is often needed to quantify the uncertainty-based information about the class variable involved in a dataset.

For many years, in order to represent the information about the class variable in classification, PT has been employed, considering that the information provided by the dataset is sufficient for precisely determining the probabilities of the class values. In the last years, classification algorithms that represent the information about the class variable via imprecise probability models have been developed. Examples can be found in [148–150]. Such algorithms consider that the information involved in a classification dataset is useful for approximating the probability of each class value, but it is not sufficient for precisely determining it. Via experimental studies, it has been shown that classification methods based on imprecise probabilities significantly outperform the ones that utilize PT when data contain class noise.[2]

Classifiers often aim to minimize the number of misclassifications. This point is optimal when all classification errors have the same importance. However, in practical applications, different classification errors usually yield different costs. For instance, in *medical diagnosis*, the consequences of incorrectly predicting that a patient does not have a serious disease might be far worse than the consequences of erroneously predicting that the patient does not have

[2] In classification, the term ‘noise’ is used to refer to errors in the data.

such a disease [140, 171, 183]; in *software defect prediction*, the cost of defective modules predicted as non-defective is probably higher than the cost of non-defective modules predicted as defective [26, 142, 191]; in *credit fraud detection*, predicting that a fraudulent credit card is legal may cause much higher economical losses for banks and financial institutions than predicting a normal credit card as fraudulent [24, 166, 182]. Thus, classifiers that take the error costs into account, known as *cost-sensitive classifiers*, have been developed.

When classifying an instance, cost-sensitive and cost-insensitive classifiers normally predict a single value of the class variable. Nevertheless, in some situations, the information provided by the dataset is not sufficient for a classifier to point out a unique class value. In these cases, it may be more logical that classifiers predict a set of values of the class variable, which is known as *Imprecise Classification* [227]. For example, suppose that, in a classification dataset for medicine, there are five class values, corresponding to five diseases that a patient can have. It is possible that, for predicting the disease of a given patient, the information provided by the dataset only allows us to know that the patient can have three of the five diseases, but there is not sufficient information for determining which of the three diseases the patient has. In this situation, even though the disease of the patient cannot be precisely determined, the information provided might be useful for knowing a suitable treatment for the patient. Nonetheless, it might be risky to predict one of the three diseases because, in case of error, the treatment could have negative consequences. Intuitively, an evaluation metric for an Imprecise Classification algorithm has to consider whether the predictions are correct (the real class value is between the predicted ones) and how informative the predictions are, which is measured by the average number of predicted class values.

In order to develop Imprecise Classification methods, imprecise probability theories are more appropriate than classical PT [227]. Few Imprecise Classification algorithms have been proposed so far. The first one was the Naïve Credal Classifier (NCC) [62, 227], which combines the IDM with the naïve assumption (all attributes are independent given the class variable) to output imprecise predictions. Afterwards, an Imprecise Classification algorithm based on Decision Trees, called the Imprecise Credal Decision Tree (ICDT), was proposed in [10]. Both NCC and ICDT were adapted for cost-sensitive classification [10].

Traditional classification (including Imprecise Classification) assumes that each instance has a unique value of a class variable. Nevertheless, in some domains, this task does not fit well since each instance may belong to multiple labels simultaneously. For example, in *text categorization*, if a text treats about the visit of Donald Trump to France, it makes sense that such a text belongs to the labels the 'United States' and 'France'; within *biology*, a protein may have multiple functions in the human body; several emotions can appear in an *image* or *music* fragment. In these domains, the *Multi-Label Classification* task (MLC) is more suitable than traditional classification. MLC aims to predict the set of labels associated with a given instance described via an attribute set.

Many approaches to MLC have been developed so far. A review of the main MLC algorithms can be seen in [97]. MLC methods can be divided into two groups. On the one hand, the *problem transformation methods* convert the MLC task into multiple traditional classification problems and then combine their solutions to provide an output for the MLC task. On the other hand, the *algorithm adaptation methods* directly adapt the existing algorithms for traditional classification to MLC. Most of these methods represent the information provided by an MLC dataset about the set of labels via classical PT. As the number of labels in MLC tends to be very high, exploiting label correlations is an important challenge for MLC algorithms. There are some approaches for determining label correlations in MLC based on precise probabilities. Moreover, in MLC, very few instances often belong to a certain label. In consequence, MLC algorithms usually suffer from a class-imbalance problem.

1.2 Objectives

In this thesis work, we follow the research lines of imprecise probability theories and models and uncertainty measures within imprecise probabilities. We also propose new classification methods based on imprecise probability models that outperform the ones of the state-of-the-art.

There are five main aims of this thesis work, which can be divided into specific objectives:

1. Firstly, we aim to analyze the properties and relations between some imprecise probability theories and models. This objective is divided into two aims:

    a) Characterize the credal sets representable through belief functions and reachable probability intervals, two imprecise probability theories such that any of them generalizes the other. For this purpose, our goal is to give a set of necessary and sufficient conditions under which a reachable set of probability intervals is representable by a belief function, as well as a characterization of belief functions representable via reachable probability intervals.

   b) Analyze the main properties of credal sets associated with the A-NPI-M, comparing them with the properties of IDM credal sets.

2. With regard to uncertainty measures, our main goal is to analyze the properties of some uncertainty measures in imprecise probability theories and models and propose uncertainty measures within imprecise probabilities that present some advantages over the existing ones. Specifically, there are four objectives concerning uncertainty measures:

   a) Make a critical analysis of recent alternatives to the maximum entropy in ET via its properties and behaviors.

   b) Study the essential mathematical properties and behavioral requirements for uncertainty measures on belief intervals for singletons. We also aim to analyze which of these properties and behavioral requirements are satisfied by each one of the uncertainty measures on belief intervals for singletons proposed so far.

   c) Introduce an uncertainty measure on belief intervals for singletons that, unlike the ones proposed so far, satisfies all the fundamental mathematical properties and behavioral requirements for this type of measure. Furthermore, we aim that, in practical applications, the proposed measure is easier to manage than the maximum entropy on a BPA, the well-established uncertainty measure in ET.

   d) Propose procedures to compute the main uncertainty measures on A-NPI-M credal sets.

3. We aim to develop a traditional classification method based on imprecise probability models that achieves better results than the existing versions of this algorithm based on classical PT, especially with class noise in the data.

4. Concerning Imprecise Classification, we aim to develop improvements over the Imprecise Classification algorithms proposed so far. Specifically, the objectives regarding Imprecise Classification can be summarized as follows:

   a) Introduce a version of the Imprecise Credal Decision Tree algorithm that uses the A-NPI-M, unlike the existing one, which utilizes the IDM. Our idea is to show that the A-NPI-M achieves statistically equivalent results to the IDM with the best choice of the parameter when both models are employed in the Imprecise Credal Decision Tree method.

  b) Propose a new version of the NCC algorithm that leads to far more informative predictions than the existing NCC.

  c) Develop the first ensemble method for Imprecise Classification. We must remark that, as imprecise classifiers tend to output a set of class values, it is not trivial to combine multiple imprecise predictions. This might be the reason why no ensemble algorithm for Imprecise Classification has been proposed so far. Therefore, in order to develop an ensemble of imprecise classifiers, we must propose a technique for combining multiple imprecise predictions.

  d) Regarding cost-sensitive Imprecise Classification, our goal is to propose a new cost-sensitive Imprecise Credal Decision Tree that presents some advantages and achieves better results than the existing one.

5. Our last goal is to introduce new MLC algorithms based on imprecise probability models that perform better than the existing ones based on precise probabilities, the improvement being more notable as there is more noise in the labels. This aim is divided into the following four objectives:

   a) Analyze the use of imprecise probabilities in two problem transformation methods for MLC, highlighting that it supposes an improvement over precise probabilities, especially with noise in the labels.

   b) Propose a new adaptation of Decision Trees for MLC that uses imprecise probabilities, unlike the existing one, which is based on classical PT. Our goal is to show that our proposed adaptation is less sensitive to noise in the labels than the one proposed so far. We aim to show, via experiments, that our proposed adaptation significantly outperforms the existing one, the improvement being more notable as there is more noise in the labels.

   c) Introduce new lazy MLC algorithms that employ imprecise probability models, unlike some lazy MLC methods developed so far, which use classical PT. Our goal is to show theoretically and experimentally that our proposed lazy MLC methods are more suitable than the existing lazy MLC algorithms based on precise probabilities to handle the class-imbalance problem that frequently appears in MLC, especially with noise in the labels.

   d) Propose a new method to exploit label correlations in MLC based on imprecise probability models. We aim to show that our proposed method has some advantages over other existing algorithms for exploiting label correlations in MLC based on classical PT. The idea is to corroborate this issue via an experimental analysis.

Finally, we also aim to apply imprecise probability models to some important domains such as *credit risk analysis* and *traffic accident analysis* to extract useful knowledge in such domains.

1.3 Organization of this thesis

This thesis work is structured into four parts plus an appendix. Each part is subdivided into chapters.

- In the first part, we contextualize our work and establish our main aims (Chapter 1).
- Part ii describes the previous knowledge necessary for our thesis work. This part is divided into five chapters. In Chapter 2, the main imprecise probability theories and models used in this work are described. Chapter 3 provides an overview of the main uncertainty measures on imprecise probabilities proposed so far. In Chapter 4, we expose the classification task, as well as the classification approaches considered in this thesis work. The Imprecise Classification task and the methods proposed so far for such a task are detailed in Chapter 5. Chapter 6 describes the Multi-Label Classification task and the main approaches to this field.
- The contributions of this thesis work are presented in Part iii. Such a part is divided into five chapters. Some imprecise probability theories and models are analyzed in Chapter 7, which corresponds to the first objective. In Chapter 8, which is associated with the second aim, we analyze some uncertainty measures within imprecise probability theories and models and propose uncertainty measures in such theories and models. A new traditional classification method based on imprecise probabilities is presented in Chapter 9. Such a chapter is associated with the third objective. Chapter 10, corresponding to the fourth aim, details our proposed Imprecise Classification algorithms. Our proposed Multi-Label Classification methods based on imprecise probability models are introduced in Chapter 11, which is associated with the fifth objective.
- Conclusions and ideas for future research are given in Part iv (Chapter 12).

Finally, in Appendix A, we show the application of some imprecise probability models to extract useful knowledge in some important domains, such as credit risk analysis and traffic accident analysis.

1.4 Contributions

The content of Part iii, which corresponds to the contributions of this thesis work, has appeared previously in the following publications, divided into journal articles and conference papers:

1.4.1 Articles

Serafín Moral-García, Joaquín Abellán, Tahani Coolen-Maturi, and Frank P.A. Coolen. "A cost-sensitive Imprecise Credal Decision Tree based on Nonparametric Predictive Inference". In: *Applied Soft Computing* 123 (2022), pp. 108916–108927. ISSN: 1568-4946.
DOI: 10.1016/j.asoc.2022.108916.

Serafín Moral-García, Javier G. Castellano, Carlos J. Mantas, and Joaquín Abellán. "Using extreme prior probabilities on the Naive Credal Classifier". In: *Knowledge-Based Systems* 237 (2022), p. 107707. ISSN: 0950-7051.
DOI: 10.1016/j.knosys.2021.107707.

Serafín Moral-García, Carlos J. Mantas, Javier G. Castellano, and Joaquín Abellán. "Using Credal C4.5 for Calibrated Label Ranking in Multi-Label Classification". In: *International Journal of Approximate Reasoning* 147 (2022), pp. 60–77. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2022.05.005.

Serafín. Moral-García, Javier G. Castellano, Carlos J. Mantas, and Joaquín Abellán. "A new label ordering method in Classifier Chains based on imprecise probabilities". In: *Neurocomputing* 487 (2022), pp. 34–45. ISSN: 0925-2312.
DOI: 10.1016/j.neucom.2022.02.048.

Serafín Moral-García Javier G. Castellano, María D. Benítez Carlos J. Mantas, and Joaquín Abellán. "A Decision Support Tool for Credit Domains: Bayesian Network with a Variable Selector Based on Imprecise Probabilities". In: *International Journal of Fuzzy Systems* 23 (2021), pp. 2004–2020.
DOI: 10.1007/s40815-021-01079-w.

Serafín Moral-García and Joaquín Abellán. “Credal sets representable by reachable probability intervals and belief functions”. In: *International Journal of Approximate Reasoning* 129 (2021), pp. 84–102. issn: 0888-613X.
doi: 10.1016/j.ijar.2020.11.007.

Serafín Moral-García and Joaquín Abellán. “Required mathematical properties and behaviors of uncertainty measures on belief intervals”. In: *International Journal of Intelligent Systems* 36.8 (2021), pp. 1–24.
doi: 10.1002/int.22432.

Serafín Moral-García and Joaquín Abellán. “Uncertainty-based information measures on the approximate non-parametric predictive inference model”. In: *International Journal of General Systems* 50.2 (2021), pp. 159–181.
doi: 10.1080/03081079.2020.1866567.

Javier G. Castellano, Serafín Moral-García, Carlos J. Mantas, and Joaquín Abellán. “On the Use of m-Probability-Estimation and Imprecise Probabilities in the Naïve Bayes Classifier”. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 28.04 (2020), pp. 661–682.
doi: 10.1142/S0218488520500282.

Serafín Moral-García and Joaquín Abellán. “Critique of modified Deng entropies under the evidence theory”. In: *Chaos, Solitons & Fractals* 140 (2020), pp. 110112–110117. issn: 0960-0779.
doi: 10.1016/j.chaos.2020.110112.

Serafín Moral-García and Joaquín Abellán. “Maximum of Entropy for Belief Intervals Under Evidence Theory”. In: *IEEE Access* 8 (2020), pp. 118017–118029.
doi: 10.1109/ACCESS.2020.3003715.

Serafín Moral-García, Carlos J. Mantas, Javier G. Castellano, and Joaquín Abellán. “Non-parametric predictive inference for solving multi-label classification”. In: *Applied Soft Computing* 88 (2020), pp. 106011–106025. issn: 1568-4946.
doi: 10.1016/j.asoc.2019.106011.

Serafín. Moral-García, Carlos J. Mantas, Javier G. Castellano, María D. Benítez, and Joaquín Abellán. “Bagging of credal decision trees for imprecise classification”. In: *Expert Systems with Applications* 141 (2020), pp. 112944–112952. issn: 0957-4174.
doi: 10.1016/j.eswa.2019.112944.

Serafín Moral-García, Javier G. Castellano, Carlos J. Mantas, Alfonso Montella, and Joaquín Abellán. "Decision Tree Ensemble Method for Analyzing Traffic Accidents of Novice Drivers in Urban Areas". In: *Entropy* 21.4 (2019). ISSN: 1099-4300.
DOI: 10.3390/e21040360.

Serafín Moral-García, Carlos J Mantas, Javier G Castellano, and Joaquín Abellán. "Using credal-C4. 5 with binary relevance for multi-label classification". In: *Journal of Intelligent & Fuzzy Systems* 35.6 (2018), pp. 6501–6512.
DOI: 10.3233/JIFS-18746.

1.4.2 Conferences

Serafín Moral García and Joaquín Abellán. "Basic Probability Assignments Representable via Belief Intervals for Singletons in Dempster-Shafer Theory". In: *Proceedings of the Twelveth International Symposium on Imprecise Probability: Theories and Applications*. Ed. by Andrés Cano, Jasper De Bock, Enrique Miranda, and Serafín Moral. Vol. 147. Proceedings of Machine Learning Research. PMLR, 2021, pp. 229–234.

Serafín Moral García, Javier García Castellano, Carlos J. Mantas Ruiz, and Joaquín Abellán. "Using Credal C4.5 for Calibrated Label Ranking in Multi-Label Classification". In: *Proceedings of the Twelveth International Symposium on Imprecise Probability: Theories and Applications*. Ed. by Andrés Cano, Jasper De Bock, Enrique Miranda, and Serafín Moral. Vol. 147. Proceedings of Machine Learning Research. PMLR, 2021, pp. 220–228.

Serafín Moral-García, Carlos J. Mantas, Javier G. Castellano, and Joaquín Abellán. "Imprecise Classification with Non-parametric Predictive Inference". In: *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Ed. by Marie-Jeanne Lesot, Susana Vieira, Marek Z. Reformat, João Paulo Carvalho, Anna Wilbik, Bernadette Bouchon-Meunier, and Ronald R. Yager. Springer International Publishing, 2020, pp. 53–66. ISBN: 978-3-030-50143-3.
DOI: 10.1007/978-3-030-50143-3_5.

Part II

BACKGROUND

2 IMPRECISE PROBABILITIES

2.1 Introduction

For making decisions, we usually need to represent the probabilistic knowledge about a finite set of possible alternatives provided by an expert or a set of observations about that set. Classical probability theory (PT) is the standard way of representing such knowledge. In many cases, this representation might be suitable. Nevertheless, in a large number of situations, a unique probability distribution may not be sufficient to represent the probabilistic knowledge involved in the set because there is not enough information available to precisely determine the probability of each alternative. For this reason, mathematical theories based on *imprecise probabilities* have been developed in the literature. All these theories generalize PT and provide some information about the probability of each alternative without determining it precisely.

Some imprecise probability theories are more suitable than others in specific situations, depending on the source of information about the set of alternatives. We must remark that, within theories based on imprecise probabilities, some of them are subsumed into others, and there are also pairs of imprecise probability theories such that any of them generalizes the other.[1] We show below the imprecise probabilities theories considered in this thesis work. Each one of these theories has specific mathematical properties.

1. **Credal sets** [137]: In this theory, the probabilistic knowledge is represented via a closed and convex set of probability distributions, also called credal set. Such a set is composed of all probability distributions compatible with the information available about the set of alternatives.

2. **Lower and upper probabilities** [195]: It uses a lower probability function and an upper probability function to determine, respectively, for each subset of alternatives, the lower and upper probability that the true alternative belongs to such a subset.

[1] The most general imprecise probability theory is coherent *lower* and *upper* previsions [208], but we do not consider it in this work thesis.

3. **Dempster-Shafer theory** or **Evidence theory** [74, 186]: It represents the probabilistic knowledge about the set of alternatives via a *basic probability assignment*, which assigns a mass probability to each subset of alternatives. Alternatively, in this theory, the probabilistic knowledge can be represented via a *belief function*, a well-known type of lower probability function. Evidence theory has been successfully used in the literature to deal with uncertainty-based information in practical applications such as *statistical classification* [78], *target identification* [41], *medical diagnosis* [34], or *face recognition* [120].

4. **Probability intervals** [70]: This theory employs a probability interval for each alternative whose lower and upper bounds indicate, respectively, the lower and upper probability of that alternative. Probability intervals have high expressive power and can be efficiently computed. For these reasons, this theory has been commonly used in practical applications such as classification [3, 11, 13, 150].

We describe these theories in detail in Section 2.2.

Moreover, in Section 2.3, we describe two imprecise probability models that allow us to make probabilistic inferences about a finite set of possible alternatives (or discrete variable) based on a set of observations about that set (or variable): The *Imprecise Dirichlet Model* [209] and the *Non-Parametric Predictive Inference Model for multinomial data* [58, 59]. These models are based on some of the aforementioned imprecise probability theories.

2.2 Imprecise probability theories

Let $X = \{x_1, \ldots, x_t\}$ be a finite set of possible alternatives[2] and $\wp(X)$ the power set of X. Let $\mathcal{P}(X)$ denote the set of all probability distributions on X.

For each probability distribution p, there is an associated probability measure that will be denoted by uppercase P.

2.2.1 Credal sets

Before defining a credal set, some previous concepts are necessary.

[2] or, alternatively, a discrete variable whose set of possible values is $\{x_1, \ldots, x_t\}$.

Given two probability distributions on X p and q, the Euclidean distance between such probability distributions can be considered:

$$\mathrm{dist}(p,q) = \sqrt{\sum_{i=1}^{t} \left(p\left(x_i\right) - q\left(x_i\right)\right)^2}. \tag{2.1}$$

Taking this distance function as a reference, for each probability distribution p on X and $\epsilon \in \mathbb{R}^+$, we can consider the ball with center p and radius ϵ, that is, the set of probability distributions on X whose distance between p and them is lower than ϵ:

$$B(p,\epsilon) = \{q \in \mathcal{P}(X) \mid \mathrm{dist}(p,q) < \epsilon\}. \tag{2.2}$$

Let us consider now a set of probability distributions on X $\mathcal{P}$.

The frontier of $\mathcal{P}$ can be defined as the set of probability distributions such that there is no ball centered on that probability distribution contained in $\mathcal{P}$ or its complement:

$$\mathrm{Fr}(\mathcal{P}) = \left\{p \in \mathcal{P} \mid B(p,\epsilon) \cap \mathcal{P} \neq \emptyset \wedge B(p,\epsilon) \cap \overline{\mathcal{P}} \neq \emptyset, \quad \forall \epsilon \in \mathbb{R}^+\right\}, \tag{2.3}$$

where $\overline{\mathcal{P}}$ denotes the complement of $\mathcal{P}$.

Definition 2.2.1 *A set of probability distributions on* X, $\mathcal{P}$, *is said to be closed if it contains its frontier, i.e,* $\mathrm{Fr}(\mathcal{P}) \subseteq \mathcal{P}$.

Also, it is said that a set of probability distributions, $\mathcal{P}$, is convex if any convex combination of probability distributions in $\mathcal{P}$ also belongs to $\mathcal{P}$:

Definition 2.2.2 *A set of probability distributions on* X, $\mathcal{P}$, *is said to be convex if* $\forall p, q \in \mathcal{P}$ *and* $\lambda \in (0,1)$, *it holds that* $\lambda p + (1-\lambda)q \in \mathcal{P}$.

Now, a *credal set* is defined in the following way [137]:

Definition 2.2.3 *A credal set on* X *is a closed and convex set of probability distributions on* X.

In a credal set, there are some probability distributions that cannot be represented as a convex combination of other probability distributions belonging to such a set. They are called the *extreme points* of the credal set.

Definition 2.2.4 *Let* $\mathcal{P}$ *be a credal set on* X. *A probability distribution* $p \in \mathcal{P}$ *is said to be an extreme point of* $\mathcal{P}$ *if* $\nexists p_1, p_2 \in \mathcal{P}$ *and* $\lambda \in (0,1)$ *such that* $p_1 \neq p_2$ *and* $p = \lambda p_1 + (1-\lambda)p_2$.

Given a credal set $\mathcal{P}$, let $\mathrm{Ext}(\mathcal{P})$ denote its set of extreme points. Though $\mathrm{Ext}(\mathcal{P})$ can be infinite, all the cases considered on this thesis will correspond to convex polytope, with a finite set of extreme points, that is, $\mathrm{Ext}(\mathcal{P}) < \infty$.

A credal set can be geometrically represented in $\mathbb{R}^t$, which can be specified through a finite set of linear constraints or via its set of extreme points. Different sets of linear constraints can lead to the same credal set. In contrast, the representation of a credal set through its set of extreme points is always unique.

For characterizing a credal set, the set of extreme points is commonly employed. Indeed, extreme points of credal sets play an important for some purposes, such as performing inference [65, 67] or computing bounds of some uncertainty measures [14]. Hence, many works have been developed in the literature for computing the set of extreme points in some types of credal sets. For example, Miranda and Destercke [156] characterized the set of extreme points of a credal set determined by comparative probabilities on singletons.

2.2.1.1 *Udpating on credal sets*

Let $\mathcal{P}$ be a credal set on X. Let us assume now that we know that the true alternative belongs to a certain subset $B \subseteq X$. The aim now is to obtain a new credal set from P with this new information, namely $\mathcal{P} \mid B$. Three cases are distinguished.

- When all the probability measures on $\mathcal{P}$ take a positive value on B, $\mathcal{P} \mid B$ is obtained by conditioning the probability distributions on $\mathcal{P}$ on B:
$$\mathcal{P} \mid B = \{p(. \mid B) \mid p \in \mathcal{P}\}. \tag{2.4}$$
- If $P(B) = 0 \quad \forall p \in \mathcal{P}$, then $\mathcal{P} \mid B$ is undetermined.
- The most interesting case arises when $\exists p_1, p_2 \in \mathcal{P}$ satisfying $p_1(B) = 0 < p_2(B)$. In this situation, there are several ways of obtaining $\mathcal{P} \mid B$. The most informative is the regular conditioning [208], which considers the probability distributions in $\mathcal{P}$ that takes a positive value on B and conditions them on B:
$$\mathcal{P} \mid B = \{p(. \mid B) \mid p \in \mathcal{P} \wedge P(B) > 0\}. \tag{2.5}$$

Definition 2.2.5 *[208] In general, the regular conditioning of $\mathcal{P}$ on B is defined as follows:*

$$\mathcal{P} \mid B = \begin{cases} \{p(. \mid B) \mid p \in \mathcal{P} \wedge P(B) > 0\} & \mathtt{if} \quad \exists p \in \mathcal{P} \mid P(B) > 0 \\ \\ \mathtt{undetermined} & \mathtt{if} \quad P(B) = 0 \quad \forall p \in \mathcal{P} \end{cases}$$

2.2.1.2 *Marginalization of credal sets*

Suppose now that we have two finite sets $X = \{x_1, x_2, \ldots, x_t\}$ and $Y = \{y_1, y_2, \ldots, y_{t'}\}$. Let $\mathcal{P}$ be a credal set on the product space $X \times Y$.

We can define the marginal credal set of $\mathcal{P}$ on X as follows:

Definition 2.2.6 *Let $\mathcal{P}$ be a credal set on $X \times Y$ and $\mathcal{P}(X)$ the set of all probability distributions on X. The set given by:*

$$\mathcal{P}^{\downarrow X} = \left\{ p_X \in \mathcal{P}(X) \mid \exists p \in \mathcal{P} : p_X\left(x_i\right) = \sum_{j=1}^{t'} p\left(x_i, y_j\right), \quad \forall i = 1, 2, \ldots, t \right\} \tag{2.6}$$

is called the marginal credal set of P *on* X.

We may note that the marginal credal set on X is composed of the marginal probability distributions on X of the probability distributions belonging to $\mathcal{P}$.

The definition of the marginal credal set of $\mathcal{P}$ on Y is analogous:

$$\mathcal{P}^{\downarrow Y} = \left\{ p_Y \in \mathcal{P}(Y) \mid \exists p \in \mathcal{P} : p_Y\left(y_j\right) = \sum_{i=1}^{t} p\left(x_i, y_j\right), \quad \forall j = 1, 2, \ldots, t' \right\}, \tag{2.7}$$

where $\mathcal{P}(Y)$ denotes the set of all probability distributions on Y.

2.2.1.3 *Independence on credal sets*

Let $X = \{x_1, x_2, \ldots, x_t\}$ and $Y = \{y_1, y_2, \ldots, y_{t'}\}$ be two finite sets. Let p be a probability distribution on the product space $X \times Y$. In classical PT, the *stochastic independence* is the standard definition of independence.

Definition 2.2.7 *It is said that* p *verifies stochastic independence when*

$$p(x_i, y_j) = p^{\downarrow X}\left(x_i\right) \times p^{\downarrow Y}\left(y_j\right), \quad \forall i = 1, 2, \ldots, t, \quad j = 1, 2, \ldots, t', \tag{2.8}$$

where $p^{\downarrow X}$ *and* $p^{\downarrow Y}$ *are the marginal probability distributions of* p *on* X *and* Y*, respectively.*

The definition of independence is quite simple in classical probability theory. Nevertheless, when the probabilistic knowledge about X and Y is given through a credal set, the concept of independence is much more complicated. Indeed, six definitions of independence on credal sets have been proposed. They are described in detail in [63].

Between these concepts, the *strong independence* is one of the most utilized in practice [66, 68, 225]. There are alternative definitions of independence in imprecise probabilities. For more details, see [63, 71].

Definition 2.2.8 *Let $\mathcal{P}$ be a credal set on $X \times Y$. Let $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$ denote, respectively, the marginal credal sets of $\mathcal{P}$ on X and Y, determined via Equations (2.6) and (2.7). It is said that there is strong independence under $\mathcal{P}$ if $\mathcal{P}$ is the convex hull of the set of probability distributions resulting from making product probabilities on the marginal credal sets. Formally:*

$$\mathcal{P} = \mathrm{CH}\left(\mathcal{P}^{\downarrow X} \times \mathcal{P}^{\downarrow Y}\right), \tag{2.9}$$

where

$$\mathcal{P}^{\downarrow X} \times \mathcal{P}^{\downarrow Y} = \left\{p_X \times p_Y \mid p_X \in \mathcal{P}^{\downarrow X} \wedge p_Y \in \mathcal{P}^{\downarrow Y}\right\} \tag{2.10}$$

and CH *denotes the convex hull of a set of probability distributions.*

2.2.2 Coherent lower and upper probability functions

Definition 2.2.9 *[55] A capacity is a mapping $\underline{P} : \wp(X) \to [0,1]$ that satisfies the following conditions:*

$\underline{P}(\emptyset) = 0$ *and* $\underline{P}(X) = 1$ *(normalization)*

$\underline{P}(A) \leqslant \underline{P}(B), \quad \forall A, B \subseteq X$ *such that* $A \subseteq B$ *(monotonicity)*

Then, a *lower probability function* is a capacity whose values are interpreted as lower bounds of probability. Likewise, *an upper probability function* is a capacity whose values are interpreted as upper bounds of probability [195].

Given a lower probability function $\underline{P}$, it is possible to define an upper probability function in the following way:

$$\overline{P}(A) = 1 - \underline{P}(\overline{A}), \tag{2.11}$$

where $\overline{A}$ is the complement of A, $\quad \forall A \subseteq X$.

Definition 2.2.10 *The upper probability function given in Equation (2.11) is called the dual or conjugate of $\underline{P}$.*

Given a lower probability function and an upper probability function, for each subset $A \subseteq X$, the lower probability of A can be interpreted as the maximum price that you are willing to pay for the subset A, supposing that you receive 1 unit if the real alternative is in A. Analogously, the upper probability of A can be interpreted as your minimum selling price of a gamble that pays you 1 unit if the true alternative belongs to A [208].

The set of probability distributions compatible with a lower probability function $\underline{P}$ (really, credal set) is determined as follows:

$$\mathcal{P}\left(\underline{P}\right) = \{p \in \mathcal{P}(X) \mid \underline{P}(A) \leqslant P(A), \quad \forall A \subseteq X\}. \tag{2.12}$$

The first condition that needs to be imposed is $\mathcal{P}\left(\underline{P}\right) \neq \emptyset$. When a lower probability function verifies such a condition it is said that it *avoids sure loss* [208].

An essential concept for lower and upper probability functions is *coherence* [208]. This concept means that a lower probability function provides realistic probability bounds for betting in the sense that they can not be improved according to the available information. In this way, a lower probability function is coherent if, for each subset, there is a probability distribution on X with the same value on that subset as the lower probability function and with a value greater or equal on the remaining subsets. Analogously for an upper probability function. Formally:

Definition 2.2.11 *A lower probability function* $\underline{P} : \wp(X) \to [0,1]$ *is said to be coherent if,* $\forall A \in \wp(X), \exists p_A \in \mathcal{P}(X)$ *such that* $P_A(A) = \underline{P}(A)$ *and* $P_A(B) \geqslant \underline{P}(B) \quad \forall B \subseteq X$ *with* $B \neq A$.

Likewise, an upper probability function $\overline{P} : \wp(X) \to [0,1]$ *is said to be coherent if,* $\forall A \in \wp(X), \exists p_A \in \mathcal{P}(X)$ *such that* $P_A(A) = \overline{P}(A)$ *and* $P_A(B) \leqslant \overline{P}(B) \quad \forall B \subseteq X$ *with* $B \neq A$.

We may note that a lower probability function $\underline{P}$ is coherent if, and only if,

$$\underline{P}(A) = \inf_{p \in \mathcal{P}(\underline{P})} P(A), \quad \forall A \subseteq X. \tag{2.13}$$

If $\underline{P}$ is a not-coherent lower probability function and $\mathcal{P}\left(\underline{P}\right) \neq \emptyset$ is the credal set associated with $\underline{P}$, then it is possible to transform $\underline{P}$ into a coherent lower probability function. Such a transformation is defined in the following way:

$$E^{\underline{P}}(A) = \inf_{p \in \mathcal{P}(\underline{P})} P(A), \quad \forall A \subseteq X. \tag{2.14}$$

Definition 2.2.12 *The set function given by Equation (2.14) is called the natural extension of* $\underline{P}$.

Given a coherent lower probability function $\underline{P}$ and its conjugate coherent upper probability function $\overline{P}$, the following properties hold [208]:

1. $\underline{P}(A) \leqslant \overline{P}(A), \quad \forall A \subseteq X.$

2. $\sum_{i=1}^{t} \underline{P}\left(\{x_i\}\right) \leqslant 1 \leqslant \sum_{i=1}^{t} \overline{P}\left(\{x_i\}\right)$.

3. $\underline{P}$ is superadditive and $\overline{P}$ is subadditive, that is, it is satisfied that:

$$\begin{aligned}&\underline{P}(A \cup B) \geqslant \underline{P}(A) + \underline{P}(B)\\&\overline{P}(A \cup B) \leqslant \overline{P}(A) + \overline{P}(B), \quad \forall A, B \subseteq X \mid A \cap B = \emptyset.\end{aligned}$$

4.
$$\underline{P}(A)+\underline{P}(B)\leqslant 1+\underline{P}(A\cap B),\quad \forall A,B\subseteq X.$$
If $A\cup B=X$, then
$$\overline{P}(A)+\overline{P}(B)\geqslant 1+\overline{P}(A\cap B),\quad \forall A,B\subseteq X.$$

5. For each $A,B\subseteq X$:
$$\begin{aligned}
&\underline{P}(A)+\underline{P}(B)\leqslant \underline{P}(A\cup B)+\overline{P}(A\cap B)\leqslant \overline{P}(A)+\overline{P}(B),\\
&\underline{P}(A)+\underline{P}(B)\leqslant \overline{P}(A\cup B)+\underline{P}(A\cap B)\leqslant \overline{P}(A)+\overline{P}(B),\\
&\underline{P}(A\cup B)+\underline{P}(A\cap B)\leqslant \underline{P}(A)+\overline{P}(B)\leqslant \overline{P}(A\cup B)+\overline{P}(A\cap B).
\end{aligned}$$

It can be observed that $\underline{P}$ and its conjugate $\overline{P}$ capture the same probabilistic knowledge about X. In consequence, $\underline{P}$ is sufficient for representing such knowledge and tends to be used for this purpose.

Suppose now that there is a credal set on X, $\mathcal{P}$. Coherent lower and upper probability functions can be extracted from $\mathcal{P}$ as follows:

$$\underline{P}(A)=\inf_{p\in\mathcal{P}}\sum_{x_i\in A}p\left(x_i\right),\quad \overline{P}(A)=\sup_{p\in\mathcal{P}}\sum_{x_i\in A}p\left(x_i\right),\quad \forall A\subseteq X. \tag{2.15}$$

If $\mathcal{P}\left(\underline{P}\right)$ is the credal set consistent with the lower probability function defined in Equation (2.15), which is determined via Equation (2.12), then it holds that $\mathcal{P}\subseteq\mathcal{P}\left(\underline{P}\right)$. However, it is not always satisfied that $\mathcal{P}\left(\underline{P}\right)\subseteq\mathcal{P}$ [208]. Thereby, the coherent lower probability function involves less probabilistic knowledge about X than the original credal set $\mathcal{P}$.

Any coherent lower probability function $\underline{P}$ is uniquely represented by a set-valued function m, which satisfies:

- $m(\emptyset)=0$,
- $\sum_{A\in\wp(X)}m(A)=1$.

This function is obtained via the Möbius transform [51, 104]:

$$m\left(A\right)=\sum_{B\subseteq A}(-1)^{|A\setminus B|}\,\underline{P}\left(B\right)\quad \forall A\subseteq X. \tag{2.16}$$

Definition 2.2.13 *The function m is known as the Möbius inverse or mass function of $\underline{P}$.*

Definition 2.2.14 *If $A\subseteq X$ satisfies $m(A)\neq 0$, it is said that A is a focal element of m.*

The inverse transformation is calculated as follows:

$$\underline{P}(A) = \sum_{B \subseteq A} m(B), \quad \forall A \subseteq X. \tag{2.17}$$

If $\overline{P}$ is the coherent upper probability function dual of $\underline{P}$, then it holds that:

$$\overline{P}(A) = \sum_{B | B \cap A \neq \emptyset} m(B), \quad \forall A \subseteq X. \tag{2.18}$$

2.2.2.1 *Choquet capacities*

Choquet capacities of order k [55], also called *k-monotone capacities*, with $k \geqslant 2$, are particular cases of coherent lower and upper probability functions.

Definition 2.2.15 *Let $\underline{P}$ be a lower probability function on X. $\underline{P}$ is said to be a Choquet capacity of order k if it satisfies that*

$$\underline{P}\left(\cup_{j=1}^{k} A_j\right) \geqslant \sum_{I_j \subseteq N_k} (-1)^{|I_j|+1} \underline{P}\left(\cap_{j \in I_j} A_j\right), \tag{2.19}$$

where $N_k = \{1, 2, \ldots, k\}$ and $A_j \subseteq X, \quad \forall j = 1, 2, \ldots, k$.

Choquet capacities of order 2 are always coherent lower probability functions [117].

It is easy to deduce that, if $\underline{P}$ is a Choquet capacity of order k and $\overline{P}$ its conjugate coherent upper probability function, then, due to duality:

$$\overline{P}\left(\cap_{j=1}^{k} A_j\right) \leqslant \sum_{I_j \subseteq N_k} (-1)^{|I_j|+1} \overline{P}\left(\cup_{j \in I_j} A_j\right).$$

Choquet capacities of order k can be characterized via the Möbius inverse. The following result, proved in [51], shows the necessary and sufficient condition that the Möbius inverse of a coherent lower probability function has to satisfy to be a Choquet capacity of order k:

Proposition 2.2.1 *Let $\underline{P}$ be a coherent lower probability function on X and m its Möbius inverse. It holds that $\underline{P}$ is a Choquet capacity or order k if, and only if,*

$$\sum_{B \subseteq \cup_{i=1}^{k} A_i, B \not\subseteq A_i \forall i} m(B) \geqslant 0, \quad \forall A_i \subset X, \quad 1 \leqslant i \leqslant k.$$

As a consequence of the previous result, if a lower probability function is a Choquet capacity of order k, then the Möbius inverse of a subset with cardinality lower or equal than k is not lower than 0. In addition, a lower probability function is a Choquet capacity or order infinity if, and only if, its corresponding Möbius inverse is non-negative. These two issues are expressed in the following corollary [51]:

Corollary 2.2.1 *Let $\overline{P}$ be a coherent lower probability on X and m its associated Möbius inverse. It is satisfied that:*

1. *If $\overline{P}$ is a Choquet capacity of order k, then $m(A) \geqslant 0, \quad \forall A \subseteq X$ such that $|A| \leqslant k$.*
2. *$\overline{P}$ is a Choquet capacity of order infinity if, and only if $m(A) \geqslant 0, \quad \forall A \subseteq X$.*

Obviously, if $k' > k$, then the theory corresponding to Choquet capacities of order k is more general than the theory associated with Choquet capacities of order k'. Therefore, the most general of all these theories is the one based on Choquet capacities of order 2.

Credal sets associated with Choquet capacities of order 2 are easier to handle than arbitrary credal sets in terms of extreme points. According to the results proved in [72], if $\underline{P}$ is a Choquet capacity of order 2 on X and $\mathcal{P}\left(\underline{P}\right)$ the credal set compatible with it, then there is a correspondence between the permutations of the elements of X and the extreme points of $\mathcal{P}\left(\underline{P}\right)$. Specifically, each permutation σ of $\{1, 2, \ldots, t\}$ gives rise to an extreme point p^{σ} of $\mathcal{P}\left(\underline{P}\right)$, determined as follows:

$$p^{\sigma}\left(x_{\sigma(i)}\right) = \underline{P}\left(\left\{x_{\sigma(i)}, \ldots, x_{\sigma(t)}\right\}\right) - \underline{P}\left(\left\{x_{\sigma(i+1)}, \ldots, x_{\sigma(t)}\right\}\right), \forall i = 1, \ldots, t. \tag{2.20}$$

We must remark that it is possible that two distinct permutations lead to the same extreme point, i.e, $p^{\sigma_1} = p^{\sigma_2}$, σ_1 and σ_2 being two permutations of $\{1, 2, \ldots, t\}$ such that $\sigma_1 \neq \sigma_2$. The maximal number of extreme points of $\mathcal{P}\left(\underline{P}\right)$ is equal to $t!$, the number of permutations of $\{1, 2, \ldots, t\}$.

2.2.3 Updating coherent lower probability functions

Let $\underline{P}$ be a coherent lower probability function on X and $\overline{P}$ its conjugate coherent upper probability function. Suppose that it is known that the true alternative belongs to a certain subset $B \subseteq X$. Now, the goal is to determine from $\underline{P}$ an updated lower probability function on X with this new information.

We consider the credal set associated with $\underline{P}$, $\mathcal{P}(\underline{P})$, computed by means of Equation (2.12). Let $\mathcal{P}(\underline{P}) \mid B$ denoted the conditional credal set of $\mathcal{P}(\underline{P})$ on B obtained via regular conditioning (Definition 2.2.5). Our aim is to determine a lower probability function from $\mathcal{P}(\underline{P}) \mid B$.

Since $\mathcal{P}(\underline{P}) \mid B$ is undetermined when $P(B) = 0 \forall p \in \mathcal{P}(\underline{P})$, if $\overline{P} = 0$, then the conditional lower probability function of $\underline{P}$ on B is also undetermined.

According to the results proved in [155, 158], the set of conditional lower probability functions on B that derive from $\mathcal{P}(\underline{P}) \mid B$ is bounded by the *regular extension*:

Definition 2.2.16 *[155, 158] The function defined, for each $A \subseteq X$, by:*

$$\underline{E}^{\underline{P}}(A \mid B) = \begin{cases} \inf_{p \in \mathcal{P}(\underline{P}) \wedge p(B) > 0} \{P(A \mid B)\} & \text{if} \quad \overline{P}(B) > 0 \\ 1 & \text{if} \quad \overline{P}(B) = 0 \wedge B \subseteq A \\ 0 & \text{if} \quad \overline{P}(B) = 0 \wedge B \not\subseteq A \end{cases} \tag{2.21}$$

is called the regular extension of $\underline{P}$ conditioned on B.

The following result, demonstrated by Miranda and Montes [157], indicates how to determine the regular extension of $\underline{P}$ given B when $\underline{P}$ is a Choquet Capacity of order 2:

Proposition 2.2.2 *Let $\underline{P}$ be a Choquet capacity of order 2 on X and $B \subseteq X$. Then, the regular extension of $\underline{P}$ conditioned on B is determined by*

$$\underline{E}^{\underline{P}}(A \mid B) = \begin{cases} \frac{\underline{P}(A \cap B)}{\underline{P}(A \cap B) + \overline{P}(\overline{A} \cap B)} & \text{if} \quad \overline{P}(\overline{A} \cap B) > 0 \\ 1 & \text{otherwise} \end{cases}$$

Moreover, $\underline{E}^{\underline{P}}(. \mid B)$ is also a Choquet capacity of order 2.

2.2.3.1 *Marginalization of coherent lower probability functions*

Let now X and Y be two finite sets and $\underline{P}$ a coherent lower probability function on the product space $X \times Y$.

Definition 2.2.17 *[43] The marginal coherent lower probability function of $\underline{P}$ on X can be defined in the following way:*

$$\underline{P}^{\downarrow X} = \underline{P}(A \times Y), \quad \forall A \subseteq X. \tag{2.22}$$

The definition of the marginal coherent lower probability function of $\underline{P}$ on Y is analogous:

$$\underline{P}^{\downarrow Y} = \underline{P}(X \times B), \quad \forall B \subseteq Y. \tag{2.23}$$

As shown in [43], for a given $k \geqslant 2$, the marginalization of a Choquet capacity of order k is also a Choquet capacity of order k. Indeed, the marginalization is a closed operation for most of the types of coherent lower probability functions [43].

2.2.4 Dempster-Shafer theory of evidence

The basis of Dempster-Shafer theory, also known as Evidence theory (ET) [74, 186], is the concept of *basic probability assignment* (BPA).

Definition 2.2.18 *A basic probability assignment is a mapping $m : \wp(X) \to [0, 1]$ such that:*

- $m(\emptyset) = 0$,
- $\sum_{A \in \wp(X)} m(A) = 1$.

For each $A \subseteq X$, the value $m(A)$ is the probability mass assigned by m to A. It expresses the degree of belief that the true alternative belongs to A but not to any particular subset of A.

Definition 2.2.19 *If $A \subseteq X$ satisfies that $m(A) > 0$, it is said that A is a focal element of m.*

Definition 2.2.20 *Let m be a BPA on X. The support of m is defined as the union of all focal elements of m:*

$$\mathrm{supp}(m) = \cup_{A \subseteq X | m(A) > 0} A.$$

When all focal elements of m are singletons, m is equivalent to a probability distribution. Consequently, the concept of BPA is an extension of the concept of probability distribution in probability theory.

A given BPA has associated with it a lower probability function and an upper probability function. They are called, respectively, belief and plausibility functions.

Definition 2.2.21 *The function given by:*

$$Bel_m(A) = \sum_{B \subseteq A} m(B), \quad \forall A \subseteq X \tag{2.24}$$

is known as the belief function corresponding to m. The function determined by:

$$Pl_m(A) = \sum_{B | A \cap B \neq \emptyset} m(B), \quad \forall A \subseteq X \tag{2.25}$$

is called the plausibility function associated with m.

For each $A \subseteq X$, $\mathrm{Bel}_m(A)$ indicates the degree of belief that the true alternative is in A or in any subset of A, whereas $\mathrm{Pl}_m(A)$ expresses the degree of belief that the real alternative is in a subset whose intersection with A is not empty. Thus, $\mathrm{Bel}_m(A)$ indicates the minimum degree of belief in A and $\mathrm{Pl}_m(A)$ the maximum degree of belief in A.

Clearly, $\mathrm{Bel}_m(A) \leqslant \mathrm{Pl}_m(A), \quad \forall A \subseteq X$.

Definition 2.2.22 *For each $A \subseteq X$, the interval $[\mathrm{Bel}_m(A), \mathrm{Pl}_m(A)]$ is called the belief interval for A.*

Belief and plausibility functions are always coherent [208]. It is easy to deduce that, $\forall A \subseteq X$,

$$\mathrm{Pl}_m(A) = 1 - \mathrm{Bel}_m(\overline{A}), \tag{2.26}$$

where $\overline{A}$ denotes the complement of A.

Hence, Pl_m is the coherent upper probability function dual or conjugate of the coherent lower probability function Bel_m.

Belief functions are Choquet capacities of order infinity. It means that, for each $k \in \mathbb{N}$ and $A_1, A_2, \ldots, A_k$, the following inequalities are satisfied:

$$\mathrm{Bel}_m\left(\cup_{j=1}^{k} A_j\right) \geqslant \sum_{I_j \subseteq N_k} (-1)^{|I_j|+1} \mathrm{Bel}_m\left(\cap_{j \in I_j} A_j\right),$$
$$\mathrm{Pl}_m\left(\cap_{j=1}^{k} A_j\right) \leqslant \sum_{I_j \subseteq N_k} (-1)^{|I_j|+1} \mathrm{Pl}_m\left(\cup_{j \in I_j} A_j\right),$$

where $N_k = \{1, 2, \ldots, k\}$.

The BPA m can be obtained from the belief function Bel_m as follows:

$$m(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \mathrm{Bel}_m(B), \quad \forall A \subseteq X. \tag{2.27}$$

In this way, m is the Möbius inverse of Bel_m. Therefore, in ET, there is a one-to-one correspondence between BPAs and belief functions. For this reason, in ET, the probabilistic knowledge about X can be expressed by a BPA, via its corresponding belief function, or through its associated plausibility function.

Alternatively, each BPA m in DST can be represented by the following function:

$$Q_m(A) = \sum_{B \supseteq A} m(B), \quad \forall A \subseteq X. \tag{2.28}$$

Definition 2.2.23 *The function determined via Equation (2.28) is called the commonality function corresponding to m.*

Again, there is a one-to-one correspondence between BPAs and commonality functions. From each commonality function, the associated BPA is determined by:

$$m(A) = \sum_{B \supseteq A} (-1)^{|B-A|} Q_m(B), \quad \forall A \subseteq X. \tag{2.29}$$

In addition, each BPA m has associated with it a probability measure known as the *pignistic transformation* of m. It is defined as follows:

$$\text{BetP}_m(A) = \sum_{B \subseteq X} m(B) \frac{|A \cap B|}{|B|}, \quad \forall A \subset X. \tag{2.30}$$

We may deduce that, for singletons, it holds that:

$$\text{BetP}_m(\{x_i\}) = \sum_{A \subseteq X | x_i \in A} \frac{m(A)}{|A|}, \quad \forall i = 1, 2, \ldots, t. \tag{2.31}$$

A given BPA m on X has associated with it the following credal set, composed of all probability distributions compatible with the corresponding belief function Bel_m:

$$\mathcal{P}_m = \{p \in \mathcal{P}(X) \mid \text{Bel}_m(A) \leqslant P(A), \quad \forall A \subseteq X\}. \tag{2.32}$$

We may note that, due to the duality relation expressed in Equation (2.26), the condition $\text{Bel}_m(A) \leqslant P(A), \quad \forall A \subseteq X$ is equivalent to $\text{Bel}_m(A) \leqslant P(A) \leqslant \text{Pl}_m(A), \quad \forall A \subseteq X$.

2.2.4.1 *Extreme points of belief functions*

Let m be a BPA on X and let Bel_m denote its associated belief function. We will say that a probability distribution is an extreme point of Bel_m if it is an extreme point of $\mathcal{P}_m$, the credal set corresponding to m.

Since belief functions are particular cases of Choquet capacities of order 2, the extreme points of belief functions can be determined as the extreme points of Choquet capacities of order 2. Thus, each permutation σ of $\{1, 2, \ldots, t\}$ gives rise to the following extreme point p_m^{σ} of Bel_m:

$$\begin{aligned} p_m^{\sigma}(x_{\sigma(i)}) &= \text{Bel}_m\left(\left\{x_{\sigma(i)}, \ldots, x_{\sigma(t)}\right\}\right) - \text{Bel}_m\left(\left\{x_{\sigma(i+1)}, \ldots, x_{\sigma(t)}\right\}\right) \\ &= \sum_{A \subseteq \left\{x_{\sigma(i)}, \ldots, x_{\sigma(t)}\right\}} m(A) - \sum_{A \subseteq \left\{x_{\sigma(i+1)}, \ldots, x_{\sigma(t)}\right\}} m(A) \\ &= \sum_{A | x_{\sigma(i)} \in A, \wedge A \cap \left\{x_{\sigma(1)}, \ldots, x_{\sigma(i-1)}\right\} = \emptyset} m(A), \quad \forall i = 1, 2, \ldots, t. \end{aligned} \tag{2.33}$$

Equation (2.33) determines the procedure to obtain the extreme point associated with the permutation σ: assign the focal elements containing $x_{\sigma(1)}$ to $p\left(x_{\sigma(1)}\right)$, remove such focal elements and iterate until reaching $x_{\sigma(t)}$ or until there are no more focal elements. The described procedure is given in Algorithm 1 [161], where $\mathcal{F}_m$ denotes the set of focal elements of m.

Algorithm 1: Procedure to compute the extreme point p_m^σ associated with the permutation σ.

```
Procedure Determine extreme point corresponding to
permutation(BPA $m$, permutation of $\{1,2,\dots,t\}$ $\sigma$)
for $i = 1$ to $t$ do
    $p_m^\sigma\left(x_{\sigma(i)}\right) \leftarrow 0$
$i \leftarrow 1$
while $\mathcal{F}_m \neq \emptyset \wedge i \leqslant t$ do
    for $A_i \in \mathcal{F}_m$ such that $x_{\sigma(i)} \in A_i$ do
        $p_m^\sigma\left(x_{\sigma(i)}\right) \leftarrow p_m^\sigma\left(x_{\sigma(i)}\right) + m(A_i)$
        $\mathcal{F}_m \leftarrow \mathcal{F}_m \setminus \{A_i\}$
    $i \leftarrow i + 1$
return $p_m^\sigma$
```

As in Choquet capacities or order 2, two distinct permutations can lead to the same extreme point, i.e, $p_m^{\sigma_1} = p_m^{\sigma_2}$, σ_1 and σ_2 being two permutations of $\{1,2,\dots,t\}$ such that $\sigma_1 \neq \sigma_2$. Again, the maximal number of extreme points of Bel_m is equal to the number of permutations of $\{1,2,\dots,t\}$, i.e, t!.

Montes and Destercke [161] proved that Bel_m has t! extreme points if, and only if, all subsets of cardinality two are focal elements of m:

Proposition 2.2.3 *Let m be a BPA on X and Bel_m its associated belief function. Then, Bel_m has t! extreme points if, and only if, $m(A) > 0 \quad \forall A \subseteq X$ such that $|A| = 2$.*

Another remarkable property, also proved by Montes and Destercke [161], is that the number of extreme points of a belief function does not decrease as the number of focal elements of the corresponding BPA increases. Formally:

Proposition 2.2.4 *Let m_1 and m_2 two BPAs on X and $\mathcal{F}_1$ and $\mathcal{F}_2$ their respective sets of focal elements. Let Bel_{m_i} denote the belief function associated with the BPA m_i, for $i = 1, 2$. If $\mathcal{F}_1 \subset \mathcal{F}_2$, then the number of extreme points of Bel_{m_2} is higher or equal than the number of extreme points of Bel_{m_1}.*

According to the previous proposition, if m_1 and m_2 are two BPAs on X such that $\mathcal{F}_1 \subset \mathcal{F}_2$, where $\mathcal{F}_i$ is the set of focal elements of m_i, for $i = 1, 2$, then the number of extreme points of m_2 is not lower than the number of extreme points of m_1. However, in this case, it is possible that both BPAs have the same number of extreme points.

2.2.4.2 *Marginalization of basic probability assignments*

Suppose now that we have two finite sets X and Y. Let m be a BPA defined on the product space $X \times Y$.

We can consider, for each $R \subseteq X \times Y$, the projections of R on X and Y:

$$R_X = \left\{x_i \in X \mid \exists y_j \in Y : (x_i, y_j) \in R\right\}, \tag{2.34}$$

$$R_Y = \left\{y_j \in Y \mid \exists x_i \in X : (x_i, y_j) \in R\right\}. \tag{2.35}$$

For each subset A of X, the sum of the mass probabilities of the subsets of $X \times Y$ whose projection on X coincides with A can be considered:

$$m^{\downarrow X}(A) = \sum_{R \mid A = R_X} m(R), \quad \forall A \subseteq X, \tag{2.36}$$

where, for each $R \subseteq X \times Y$, R_X is the projection of R on X, computed via Equation (2.34).

Definition 2.2.24 *The BPA determined by means of Equation (2.36) is called the marginal BPA of* m *on* X.

Analogously, the marginal BPA of m *on* Y, $m^{\downarrow Y}$, *can be defined:*

$$m^{\downarrow Y}(B) = \sum_{R \mid B = R_Y} m(R), \quad \forall B \subseteq Y, \tag{2.37}$$

R_Y *being the projection of* R *on* Y, $\quad \forall R \subseteq X \times Y$

When all focal elements of m are “rectangles", it is said that the marginal BPAs, $m^{\downarrow X}$ and $m^{\downarrow Y}$, are *non-interactive*. Formally:

Definition 2.2.25 *Let* m *be a BPA on* $X \times Y$ *and* $m^{\downarrow X}$ *and* $m^{\downarrow Y}$ *its marginal BPAs on* X *and* Y, *respectively. It is said that* $m^{\downarrow X}$ *and* $m^{\downarrow Y}$ *are non-interactive if it is satisfied that*

- $m(A \times B) = m^{\downarrow X}(A) m^{\downarrow Y}(B), \quad \forall A \subseteq X,\ B \subseteq Y.$
- $m(C) = 0$ *if* C *does not take the form* $C = A \times B$, *with* $A \subseteq X$, $B \subseteq Y$.

Given a BPA m on $X \times Y$, its marginal BPAs on X and Y can also be obtained by means of the corresponding marginal belief functions. Formally, let Bel_m be the belief function associated with m and $Bel_m^{\downarrow X}$ and $Bel_m^{\downarrow Y}$ the marginal belief functions of Bel_m on X and Y, determined, respectively, via Equations (2.22) and (2.23):

$$\begin{aligned} Bel_m^{\downarrow X}(A) &= Bel_m(A \times Y), \quad \forall A \subseteq X. \\ Bel_m^{\downarrow Y}(B) &= Bel_m(X \times B), \quad \forall B \subseteq Y. \end{aligned} \tag{2.38}$$

Now, the marginal BPAs of m can be computed in the following way:

$$\begin{aligned} m(A)^{\downarrow X} &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} Bel_m^{\downarrow X}(B), \quad \forall A \subseteq X. \\ m(C)^{\downarrow Y} &= \sum_{B \subseteq C} (-1)^{|C \setminus B|} Bel_m^{\downarrow Y}(B), \quad \forall C \subseteq Y. \end{aligned} \tag{2.39}$$

2.2.4.3 *Updating belief functions*

Let m be a BPA on X. Let Bel_m and Pl_m denote, respectively, the belief and plausibility functions associated with m. Suppose that we know that the true alternative belongs to a certain subset $B \subseteq X$.

Dempster's rule of conditioning [74, 186] is a very important rule in ET for updating plausibility functions. According to that rule, the plausibility function conditioned on B derived from Pl_m is defined as follows:

$$Pl_m(A \mid B) = \frac{Pl_m(A \cap B)}{Pl_m(B)}, \quad \forall A \subseteq X. \tag{2.40}$$

Nonetheless, the conditional plausibility function given by Equation (2.40) might not make sense from the perspective of conditioning of lower previsions [217]. Thus, it makes more sense to consider the regular extension of Bel_m conditioned on B, which bounds the set of conditional belief functions obtained from the credal set consistent with Bel_m conditioned on B.

Since belief functions are particular cases of Choquet capacities of order 2, the regular extension of a belief function conditioned on B can be determined through Proposition 2.2.2.

However, following the results proved by Miranda and Montes [157], in the particular case of belief functions, the regular extension can be obtained via the following proposition.

Proposition 2.2.5 *Let* m *be a BPA on* X *and* $B \subset X$. *Let* $\mathrm{supp}(m)$ *denote the support of* m *and* Bel_m *its corresponding belief function. Then, the regular extension of* Bel_m *conditioned on* B *can be computed in the following way:*

$$\underline{E}^{\mathrm{Bel}_m}(A \mid B) = \begin{cases} 1 & \text{if} \quad B \cap \mathrm{supp}(m) \subseteq A \\ \\ 0 & \quad\quad \text{otherwise} \end{cases}$$

2.2.4.4 *Belief intervals for singletons*

Given a BPA m on X, the set of belief intervals for singletons corresponding to m can be considered:

$$\mathcal{I}_m = \{[\mathrm{Bel}_m(\{x_i\}), \mathrm{Pl}_m(\{x_i\})], \quad i = 1, 2, \ldots, t\}, \tag{2.41}$$

where Bel_m and Pl_m denote, respectively, the belief and plausibility functions associated with m.

The set of intervals $\mathcal{I}_m$ leads to the following credal set [70]:

$$\mathcal{P}(\mathcal{I}_m) = \{p \in \mathcal{P}(X) \mid \mathrm{Bel}_m(\{x_i\}) \leqslant p(x_i) \leqslant \mathrm{Pl}_m(\{x_i\}), \quad \forall i = 1, 2, \ldots, t\}. \tag{2.42}$$

This set is composed of all probability distributions consistent with the belief intervals for singletons.

Clearly, if a probability distribution is compatible with Bel_m, then it belongs to $\mathcal{P}(\mathcal{I}_m)$, i.e, $\mathcal{P}_m \subseteq \mathcal{P}(\mathcal{I}_m)$.

Nonetheless, there might be some probability distributions in $\mathcal{P}(\mathcal{I}_m)$ that are not consistent with Bel_m. This point is shown in Example 2.2.1 [2].

Example 2.2.1 *Let* $X = \{x_1, x_2, x_3, x_4\}$ *be a finite set. We consider the following BPA* m *on* X*:*

$$m(\{x_1, x_2\}) = 0.5, \quad m(\{x_3, x_4\}) = 0.5.$$

Let Bel_m *and* Pl_m *denote, respectively, the belief and plausibility functions associated with* m.

We have the following set of belief intervals for singletons:

$$\{[\mathrm{Bel}_m(\{x_i\}), \mathrm{Pl}_m(\{x_i\})] = [0, 0.5], \quad i \in \{1, 2, 3, 4\}\}.$$

The probability distribution $p = (p(x_1), p(x_2), p(x_3), p(x_4)) = (0, 0, 0.5, 0.5)$ *is consistent with the belief intervals for singletons as* $p(x_i) \in [0, 0.5]$, $\forall i = 1, 2, 3, 4$. *However,* p *is not compatible with* m *since*

$$P(\{x_1, x_2\}) = 0 < 0.5 = \mathrm{Bel}_m(\{x_1, x_2\}).$$

Thereby, the set of belief intervals for singletons may involve less probabilistic knowledge about X than the associated BPA.

2.2.4.5 *p-boxes*

Let $X = \{x_1, x_2, \ldots, x_t\}$ be a finite set that is also totally ordered, i.e, i.e $x_1 < x_2 < \ldots < x_t$.

Definition 2.2.26 *A subset $A \subseteq X$ is said to be an interval if any element between two elements belonging to A is also in A.*

For intervals, the following notation is used: $[x_i, x_{i+k}] = \{x_i, x_{i+1}, \ldots, x_{i+k}\}$.

Definition 2.2.27 *[90] A p-box $\left(\underline{F}, \overline{F}\right)$ is a pair of functions $\underline{F}, \overline{F} : X \to [0,1]$ satisfying*

- $\underline{F}(x_t) = \overline{F}(x_t) = 1$.
- $\underline{F}(x_i) \leqslant \overline{F}(x_i), \quad \forall i = 1, 2, \ldots, t$.
- *Both $\underline{F}$ and $\overline{F}$ are increasing, i.e, $\underline{F}(x_i) \leqslant \underline{F}(x_j) \wedge \overline{F}(x_i) \leqslant \overline{F}(x_j), \quad \forall i, j \in \{1, 2, \ldots, t\}$ such that $i \leqslant j$.*

P-boxes can be interpreted as lower and upper bounds of imprecisely defined cumulative distribution functions. In consequence, the following credal set corresponds to the p-box $\left(\underline{F}, \overline{F}\right)$:

$$\mathcal{P}\left(\underline{F}, \overline{F}\right) = \left\{p \in \mathcal{P}(X) \mid \underline{F}(x_i) \leqslant F_p(x_i) = P([x_1, x_i]) \leqslant \overline{F}(x_i), \quad \forall i = 1, \ldots, t\right\}, \tag{2.43}$$

where F_p denotes the cumulative distribution function corresponding to the probability distribution p.

Each p-box $\left(\underline{F}, \overline{F}\right)$ on X corresponds to a belief function on X, which is determined in the following way [79]:

$$\mathrm{Bel}_F(A) = \min\left\{p(A) \mid \underline{F}(x_i) \leqslant F_p(x_i) \leqslant \overline{F}(x_i), \quad \forall i = 1, \ldots, t\right\}, \quad \forall A \subseteq X. \tag{2.44}$$

Belief functions equivalent to a p-box were been characterized in [79], (that is, belief functions Bel for which there exists a p-box $\left(\underline{F}, \overline{F}\right)$ such that $\mathcal{P}\left(\underline{F}, \overline{F}\right) = \mathcal{P}\left(\mathrm{Bel}\right)$, $\mathcal{P}\left(\mathrm{Bel}\right)$ being the credal set consistent with Bel).

Let Bel be a belief function on X. Let m_{Bel} be the BPA associated with Bel, which is computed via Equation (2.27).

We consider the following interval order:

$$[a_1, a_2] \preceq [b_1, b_2] \Leftrightarrow a_1 \leqslant b_1 \wedge a_2 \leqslant b_2. \tag{2.45}$$

The belief function Bel is equivalent to a p-box if, and only if, its focal elements $A_1, A_2, \ldots, A_k$ are intervals ordered with $A_1 \preceq A_2 \preceq \ldots \preceq A_k$. In that case, a subset is said to be a focal element of the p-box if it is a focal element of the corresponding belief function.

Montes and Destercke [161] developed a procedure to compute the set of extreme points of a belief function equivalent to a p-box.

2.2.4.6 *Possibility measures*

Definition 2.2.28 *[187] A possibility measure on* X *is a mapping* $\mathrm{Poss} : \wp(X) \to [0,1]$ *such that*

$$\mathrm{Poss}(A) = \max_{x_i \in A} \{\mathrm{Poss}\left(\{x_i\}\right)\}, \quad \forall A \subseteq X.$$

The function Nec *determined by*

$$\mathrm{Nec}(A) = 1 - \mathrm{Poss}(\overline{A}), \quad \forall A \subseteq X$$

is called the necessity measure corresponding to Poss*.*

A possibility measure is always a plausibility function, and its associated necessity measure is always a belief function. Consequently, possibility measures inherit all mathematical properties and characteristics of belief and plausibility functions.

As shown in [187], the focal elements of a necessity measure are always nested, that is, if $A_1, A_2, \ldots, A_k$ are the focal elements of Nec, then $A_1 \subset A_2 \subset \ldots \subset A_k$.

2.2.5 Reachable probability intervals

In probability intervals theory [70], the probabilistic knowledge about X is represented by a set of probability intervals on singletons

$$\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}, \tag{2.46}$$

where $0 \leqslant l_i \leqslant u_i \leqslant 1, \quad \forall i = 1, 2, \ldots, t$.

For each $i \in \{1, 2, \ldots, t\}$, l_i and u_i represent, respectively, the lower and upper bounds of the probability of x_i. Therefore, the set of probability intervals $\mathcal{I}$ has associated with it the following credal set, composed of the probability distributions on X compatible with such intervals [70]:

$$\mathcal{P}\left(\mathcal{I}\right) = \{p \in \mathcal{P}(X) \mid l_i \leqslant p\left(x_i\right) \leqslant u_i, \quad \forall i = 1, 2, \ldots, t\}. \tag{2.47}$$

For the credal set given by Equation (2.47) not to be empty (in such a case, there would not be any probability distribution consistent with the intervals), it is necessary to impose the following condition to the set of probability intervals $\mathcal{I}$ [70]:

Definition 2.2.29 *A given set of probability intervals* $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$ *is said to be proper if the sum of the lower bounds is lower or equal than* 1 *and the sum of the upper bounds is greater or equal than* 1*:*

$$\sum_{i=1}^{t} l_i \leqslant 1 \leqslant \sum_{i=1}^{t} u_i. \tag{2.48}$$

It is easy to deduce that $\mathcal{P}(\mathcal{I}) \neq \emptyset$, i.e, the set of intervals avoids sure loss if, and only if, the condition given in Equation (2.48) is satisfied.

Let us consider the coherent lower probability function associated with $\mathcal{P}(\mathcal{I})$:

$$\underline{P}(A) = \inf_{p \in \mathcal{P}(\mathcal{I})} P(A), \quad \forall A \subseteq X. \tag{2.49}$$

Definition 2.2.30 *The coherent lower probability function defined in Equation (2.49) is called the natural extension of* $\mathcal{I}$*.*

Let $\underline{P}$ be the coherent upper probability function conjugate of $\underline{P}$. Clearly, $\underline{P}(\{x_i\}) \geqslant l_i$ and $\overline{P}(\{x_i\}) \leqslant u_i, \quad \forall i = 1, 2, \ldots, t$. For a proper set of probability intervals to be coherent, the *reachability* condition has to be imposed [70].

Definition 2.2.31 *It is said that a proper set of probability intervals* $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$ *is reachable if*

$$\underline{P}(\{x_i\}) = l_i \wedge \overline{P}(\{x_i\}) = u_i, \quad \forall i = 1, 2, \ldots, t.$$

If the reachability condition is violated, then there might be some values of the intervals in $\mathcal{I}$ that do not correspond to any probability distribution belonging to $\mathcal{P}(\mathcal{I})$. In these situations, the intervals are unnecessarily broad.

The following result, which was demonstrated in [70], allows quickly checking the reachability of a given proper set of probability intervals:

Proposition 2.2.6 *Let* $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$ *be a proper set of probability intervals.* $\mathcal{I}$ *is reachable if, and only if,*

$$\sum_{j=1, j\neq i}^{t} u_j + l_i \geqslant 1 \geqslant \sum_{j=1, j\neq i}^{t} l_j + u_i, \quad \forall i = 1, 2, \ldots, t.$$

Given a non-reachable proper set of probability intervals $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$, it is possible to transform it into the following reachable set of probability intervals

$$\mathcal{I}' = \left\{[l_i', u_i'], \quad i = 1, 2, \ldots, t\right\},$$

where

$$l_i' = \max\left(l_i, 1 - \sum_{j=1, j\neq i}^{t} u_i\right), \quad u_i' = \min\left(u_i, 1 - \sum_{j=1, j\neq i}^{t} l_i\right), \quad \forall i = 1, 2, \ldots, t.$$

Indeed, in [70], it was shown that $\mathcal{I}'$ is reachable and the set of probability distributions compatible with $\mathcal{I}$ coincides with the set of probability distributions consistent with $\mathcal{I}'$.

The following proposition, which was proved in [70], lets us obtain the natural extension of a reachable set of probability intervals and its corresponding coherent upper probability function:

Proposition 2.2.7 *If $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$ is a reachable set of probability intervals, then its natural extension $\underline{P}$ and its associated coherent upper probability function $\overline{P}$ are determined, for each $A \subseteq X$, by:*

$$\underline{P}(A) = \max\left(\sum_{x_i \in A} l_i, 1 - \sum_{x_i \notin A} u_i\right), \quad \overline{P}(A) = \min\left(\sum_{x_i \in A} u_i, 1 - \sum_{x_i \notin A} l_i\right).$$

As demonstrated in [70], natural extensions of reachable probability intervals are Choquet capacities of order 2. It means that, if $\underline{P}$ is the natural extension of a reachable set of probability intervals and $\overline{P}$ its conjugate coherent upper probability function, then the following inequalities hold $\forall A, B \subseteq X$.

$$\underline{P}(A \cup B) + \underline{P}(A \cap B) \geqslant \underline{P}(A) + \underline{P}(B),$$
$$\overline{P}(A \cup B) + \overline{P}(A \cap B) \leqslant \overline{P}(A) + \overline{P}(B).$$

Nevertheless, the natural extension of a reachable set of probability intervals is not always a belief function (Choquet capacity of order infinity). In Example 2.2.2 [2], we show a case in which the Möbius inverse of the natural extension of a reachable set of probability intervals is not non-negative.

Example 2.2.2 *Suppose that we have a finite set $X = \{x_1, x_2, x_3\}$. Let us consider the following reachable set of probability intervals on X:*

$$\mathcal{I} = \{[0, 0.5]\,;\ [0, 0.5]\,;\ [0, 0.5]\}.$$

Let $\underline{P}$ denote the natural extension of $\mathcal{I}$ and m its Möbius inverse.
For singletons,

$$m(\{x_i\}) = \underline{P}(\{x_i\}) = l_i = 0, \quad \forall i = 1, 2, 3.$$

Concerning the subsets of cardinality two:

$$\underline{P}\left(\{x_i, x_j\}\right) = \max(l_i + l_j, 1 - u_k) = 1 - u_k = 0.5,$$
$$m\left(\{x_i, x_j\}\right) = 1 - u_k - l_i - l_j = 0.5, \quad \forall 1 \leqslant i < j \leqslant 3, k \neq i, k \neq j.$$

So,

$$\begin{aligned} m\left(\{x_1, x_2, x_3\}\right) = 1 - \sum_{A \subset X} m(A) \\ = 1 - m\left(\{x_1, x_2\}\right) - m\left(\{x_1, x_3\}\right) - m\left(\{x_2, x_3\}\right) - \\ m\left(\{x_1\}\right) - m\left(\{x_2\}\right) - m\left(\{x_3\}\right) \\ = 1 - 0.5 - 0.5 - 0.5 = -0.5 < 0. \end{aligned}$$

Reachable probability intervals are neither generalizations of belief functions. Actually, in Example 2.2.1, we have shown a situation in which a belief function and its corresponding set of belief intervals for singletons do not represent the same uncertainty-based information about X.

2.2.5.1 *Extreme points of reachable probability intervals*

The credal set compatible with a reachable set of probability intervals is often represented through linear constraints. An alternative representation of such a credal set is its set of extreme points. The set of extreme points of the credal set corresponding to a reachable set of probability intervals can be vary large. As demonstrated in [197], the maximal number of extreme points of a reachable set of probability intervals on X is equal to

$$e(t) = \begin{cases} \binom{t+1}{(t+1)/2} \times \frac{t+1}{4} & \text{if} \quad t \quad \text{is} \quad \text{odd} \\ \\ \binom{n+1}{t/2} \times \frac{t}{2} & \text{if} \quad t \quad \text{is} \quad \text{even} \end{cases} \tag{2.50}$$

Thereby, the representation via linear constraints is probably more efficient than the one corresponding to extreme points [70]. However, as explained before, the set of extreme points is usually needed for making inferences.

In [70], a recursive procedure for obtaining the set of extreme points of a reachable set of probability intervals was proposed. That algorithm maintains a global set $\text{Ext}\left(\mathcal{P}(\mathcal{I})\right)$, which contains the extreme points of $\mathcal{P}(\mathcal{I}_m)$ found in each moment. The procedure utilizes an implicit tree search in which each node corresponds to a partial probability distribution p ("partial" means that $l_i \leqslant p(x_i) \leqslant u_i \quad \forall i = 1, 2, \ldots, t$ but it might hold that $\sum_{i=1}^{t} p(x_i) < 1$). At the root node, $p\left(x_i\right) = l_i \quad \forall i = 1, 2, \ldots, t$. Each child node is a refinement of its parent node in which one component is incremented. The leaf nodes of the tree correspond to the extreme points. In each node, there are two local variables: Expl, which contains the indices i whose component cannot be incremented as $p(x_i) = u_i, \quad i \in \{1, 2, \ldots, t\}$, and a real value λ, which indicates the remaining probability mass to distribute among the components ($\lambda = 1 - \sum_{i=1}^{t} p(x_i)$).

Hence, the procedure for determining the set of extreme points of the credal set consistent with a reachable set of probability intervals is given in Algorithm 2 [70]. At the end of this procedure, $\mathrm{Ext}(\mathcal{P}(\mathcal{I}))$ contains all extreme points of $\mathcal{P}(\mathcal{I})$.

Algorithm 2: Procedure to compute the set of extreme points of $\mathcal{P}(\mathcal{I})$.

```
$\mathrm{Ext}(\mathcal{P}(\mathcal{I})) = \emptyset$
for $i = 1$ to $t$ do
    $p(x_i) = l_i$
$\mathrm{Expl} = \emptyset$
$\lambda = 1 - \sum_{i=1}^{t} l_i$
GetExtremePoints(p,$\lambda$,Expl)
for $i = 1$ to $t$ do
    if $i \notin \mathrm{Expl}$ then
        $aux \leftarrow p(x_i)$
        if $\lambda \leqslant u_i$ then
            $p(x_i) \leftarrow p(x_i) + \lambda$
            if $p \notin \mathrm{Ext}(\mathcal{P}(\mathcal{I}))$ then
                $\mathrm{Ext}(\mathcal{P}(\mathcal{I}) \leftarrow \mathrm{Ext}(\mathcal{P}(\mathcal{I})) \cup \{p\}$
        else
            $p(x_i) \leftarrow u_i$
            $\lambda' \leftarrow \lambda + u_i - aux$
            GetExtremePoints(p, $\lambda'$,Expl$\cup\{i\}$)
        $p(x_i) \leftarrow aux$
```

2.2.6 Marginalization of reachable probability intervals

Suppose now that we have two finite sets $X = \{x_1, x_2, \ldots, x_t\}$ and $Y = \{y_1, y_2, \ldots, y_{t'}\}$. Let $\mathcal{I} = \left\{[l_{ij}, u_{ij}], \quad i = 1, 2, \ldots, t, \quad j = 1, 2, \ldots, t'\right\}$ be a reachable set of probability intervals on $X \times Y$. Let $\mathcal{P}(\mathcal{I})$ denote the credal set corresponding to $\mathcal{I}$. Let $\underline{P}$ be the natural extension of $\mathcal{I}$ and $\overline{P}$ its dual coherent upper probability function. Let $\underline{P}^{\downarrow X}$ and $\underline{P}^{\downarrow Y}$ denote, respectively, the marginal lower probability functions of $\underline{P}$ on X and Y. Analogously, let $\overline{P}^{\downarrow X}$ and $\overline{P}^{\downarrow Y}$ denote the marginal upper probability functions of $\overline{P}$ on X and Y, respectively.

In these cases, projecting on the credal set associated with $\mathcal{I}$ is equivalent to projecting on the natural extension or its associated coherent upper probability function. It is expressed in the following proposition [70]:

Proposition 2.2.8 *With the previous notation, it holds that*

$$\begin{aligned} \underline{P}^{\downarrow X}(A) &= \min_{p \in \mathcal{P}^{\downarrow X}} P(A), \quad \overline{P}^{\downarrow X}(A) = \max_{p \in \mathcal{P}^{\downarrow X}} P(A), \quad \forall A \subseteq X, \\ \underline{P}^{\downarrow Y}(B) &= \min_{p \in \mathcal{P}^{\downarrow Y}} P(B), \quad \overline{P}^{\downarrow Y}(B) = \max_{p \in \mathcal{P}^{\downarrow Y}} P(B), \quad \forall B \subseteq Y, \end{aligned}$$

where $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$ are the marginal credal sets of $\mathcal{P}$ on X and Y, respectively.

The following result, proved in [70], shows how to obtain the marginal reachable sets of probability intervals from the reachable set of probability intervals defined on $X \times Y$:

Proposition 2.2.9 *Let $\mathcal{I} = \left\{[l_{ij}, u_{ij}], \quad i = 1, 2, \ldots, t, \quad j = 1, 2, \ldots, t'\right\}$ be a reachable set of probability intervals on $X \times Y$ and $\underline{P}$ its natural extension. The marginal lower probability function of $\underline{P}$ on X is associated with the following reachable set of probability intervals on X*

$$\mathcal{I}^{\downarrow X} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\},$$

where

$$\begin{aligned} l_i &= \max\left(\sum_{j=1}^{t'} l_{ij}, 1 - \sum_{k \neq i} \sum_{j=1}^{t'} u_{kj}\right), \\ u_i &= \min\left(\sum_{j=1}^{t'} u_{ij}, 1 - \sum_{k \neq i} \sum_{j=1}^{t'} l_{kj}\right), \quad \forall i = 1, 2, \ldots, t. \end{aligned}$$

The determination of the marginal reachable set of probability intervals of $\mathcal{I}$ on Y is analogous.

2.2.7 Conditioning on reachable probability intervals

Let $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t,\}$ be a reachable set of probability intervals on X. Suppose that we know that the true alternative belongs to a subset

$B \subseteq X$. Let $\underline{P}$ denote the natural extension of $\mathcal{I}$ and $\overline{P}$ its conjugate coherent upper probability function. The aim is to obtain the set of probability intervals on X, $\mathcal{I} \mid B = \left\{[l_{i|B}, u_{i|B}], \quad i = 1, 2, \dots, t\right\}$, with this new information. For this purpose, we aim to obtain a conditional lower probability function from $\underline{P}$ on B, $\underline{P} \mid B$ and use the following equalities for singletons:

$$l_{i|B} = \underline{P}\left(\{x_i\} \mid B\right), \quad u_{i|B} = \overline{P}\left(\{x_i\} \mid B\right), \quad \forall i = 1, 2, \dots, t. \tag{2.51}$$

Clearly, if $\overline{P}(B) = 0$, which happens if, and only if, $u_i = 0 \quad \forall x_i \in B$, then $\mathcal{I} \mid B$ is undetermined. So, hereon, we suppose that $\sum_{x_i \in B} u_i > 0$.

Since natural extensions of reachable sets of probability intervals are always Choquet capacities of order 2, Proposition 2.2.2 can be used for obtaining $\underline{P} \mid B$. For each singleton $\{x_i\}$, $i \in \{1, 2, \dots, t\}$, we distinguish two cases:

- If $u_i = 0$ or $x_i \notin B$ then, clearly, $l_{i|B} = u_{i|B} = 0$.
- If $u_i > 0 \wedge x_i \in B$ then, $\overline{\{x_i\}} \cap B = B \setminus \{x_i\}$. If $\overline{P}(B \setminus \{x_i\}) = 0$, i.e, $\sum_{x_j \in B \setminus \{x_i\}} u_j = 0$, then $l_{i|B} = u_{i|B} = 1$. Otherwise, it holds that:

$$l_{i|B} = \underline{P}\left(\{x_i\} \mid B\right) = \frac{\underline{P}(\{x_i \cap B\})}{\underline{P}(\{x_i \cap B\}) + \overline{P}\left(\overline{\{x_i\}} \cap B\right)} = \frac{l_i}{l_i + \overline{P}\left(B \setminus \{x_i\}\right)},$$

$$u_{i|B} = \overline{P}\left(\{x_i\} \mid B\right) = 1 - \underline{P}\left(\overline{\{x_i\}} \mid B\right) = 1 - \frac{\underline{P}\left(\overline{\{x_i\}} \cap B\right)}{\underline{P}\left(\overline{\{x_i\}} \cap B\right) + \overline{P}\left(\{x_i\} \cap B\right)}$$

$$= 1 - \frac{\underline{P}(B \setminus \{x_i\})}{\underline{P}(B \setminus \{x_i\}) + u_i} = \frac{u_i}{u_i + \underline{P}\left(B \setminus \{x_i\}\right)}.$$

Moreover, as $\underline{P}$ is the natural extension of a reachable set of probability intervals, Proposition 2.2.7 let us deduce that, for each $i \in \{1, 2, \dots, t\}$:

$$\begin{aligned} \overline{P}\left(B \setminus \{x_i\}\right) &= \max\left(\sum_{x_j \in B} l_j - l_i, 1 - u_i - \sum_{x_j \notin B} u_j\right), \\ \overline{P}\left(B \setminus \{x_i\}\right) &= \min\left(\sum_{x_j \in B} u_j - u_i, 1 - l_i - \sum_{x_j \notin B} l_j\right) \end{aligned} \tag{2.52}$$

Hence, we have the following conditional set of probability intervals of $\mathcal{I}$ on B:

$$\left\{\left[l_{i|B}, u_{i|B}\right], \quad i = 1, 2, \dots, t]\right\}, \tag{2.53}$$

where

$$l_{i|B} = \begin{cases} 0 & \text{if} \quad u_i = 0 \vee x_i \notin B \\ 1 & \text{if} \quad u_i > 0,\, x_i \in B \wedge \sum_{x_j \in B\setminus\{x_i\}} u_j = 0 \\ \frac{l_i}{l_i + \overline{P}(B\setminus\{x_i\})} & \text{if} \quad u_i > 0,\, x_i \in B \wedge \sum_{x_j \in B\setminus\{x_i\}} u_j > 0 \end{cases}$$

$$u_{i|B} = \begin{cases} 0 & \text{if} \quad u_i = 0 \vee x_i \notin B \\ 1 & \text{if} \quad u_i > 0,\, x_i \in B \wedge \sum_{x_j \in B\setminus\{x_i\}} u_j = 0 \\ \frac{u_i}{u_i + \underline{P}(B\setminus\{x_i\})} & \text{if} \quad u_i > 0,\, x_i \in B \wedge \sum_{x_j \in B\setminus\{x_i\}} u_j > 0 \end{cases}$$

where $\underline{P}\,(B \setminus \{x_i\})$ and $\overline{P}\,(B \setminus \{x_i\})$ are computed by means of Equation (2.52), $\forall i = 1, 2, \ldots, t$.

As demonstrated in [157], the regular extension of a Choquet capacity of order 2 is always a Choquet capacity of order 2. Therefore, the conditional set of probability intervals on B, $\mathcal{I} \mid B$, is always reachable.

2.2.8 Relationships among imprecise probabilities theories

Among the theories based on imprecise probabilities considered in this thesis work, the most general is the one based on credal sets. In fact, in all these theories, the probabilistic knowledge can be represented via a credal set.

The second-most general theory is coherent lower probability functions. A lower probability function always determines a credal set. Nevertheless, a credal set is not always representable by a lower probability function.

Choquet capacities of order k are an important family of coherent lower probability functions. If $k' > k$, then the theory based on Choquet capacities of order k' is less general than the theory based on Choquet capacities of order k. The less general theory of Choquet capacities is the one of order infinity. Such a theory is known as Evidence theory or Dempster-Shafer theory. In it, coherent lower probability functions are called belief functions.

Reachable probability intervals are Choquet capacities of order 2. However, they are not always representable by means of a belief function. Also, reachable probabilities intervals do not extend belief functions. In consequence, belief functions and reachable probability intervals are not comparable in terms of generalities.

Figure 2.1 shows an order of the imprecise probability theories considered in this thesis work concerning generalities.

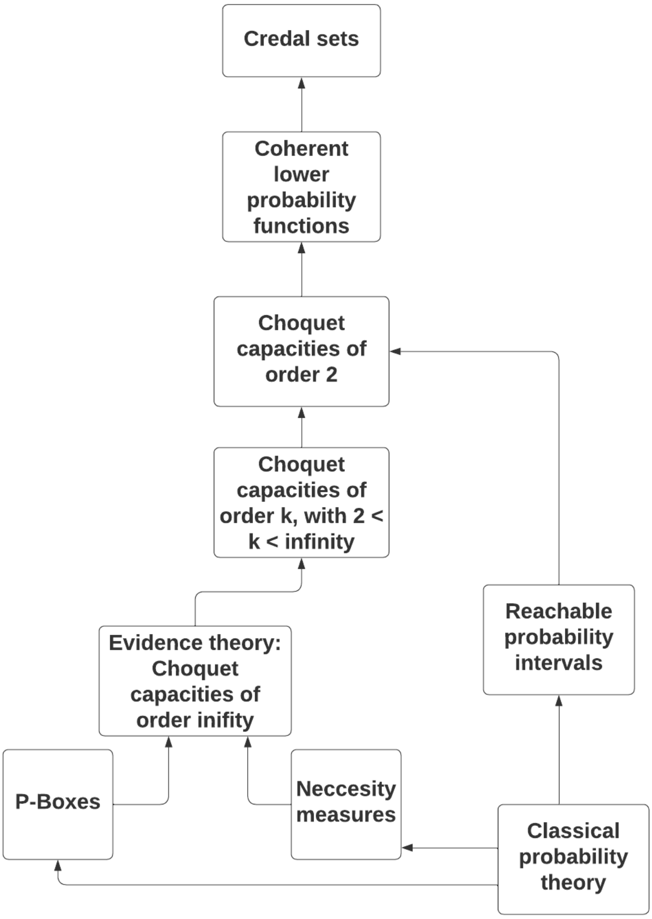
Credal sets
Coherent lower probability functions
Choquet capacities of order 2
Choquet capacities of order k, with 2 < k < infinity
Evidence theory: Choquet capacities of order inifity
Reachable probability intervals
P-Boxes
Neccesity measures
Classical probability theory

Figure 2.1: Order of the imprecise probabilities theory considered in this thesis work regarding generalities. Arrows are directed towards a more general theory.

2.3 Special models based on imprecise probabilities

Let X be a discrete variable that takes values in a finite set $\{x_1, x_2, \dots, x_t\}$[3]. Suppose that we have a sample $\mathcal{D}$ of N independent and identically distributed observations about X.

For each $i = 1, 2, \dots, t$, let $n(x_i)$ denote the number of occurrences of the x_i value in the sample. Likewise, for each $A \subseteq \{x_1, x_2, \dots, x_t\}$, let $n(A)$ denote the number of observations in the sample for which the value of X belongs to A. Let $\mathcal{P}(X)$ be the set of all probability distributions on X.

Suppose that we want to make inferences about
$p_{N+1} = (p_{N+1}(x_1), p_{N+1}(x_2), \dots, p_{N+1}(x_t))$, where $p_{N+1}(x_i)$ indicates the probability that the next observation takes the x_i value, $\forall i = 1, 2, \dots, t$.

2.3.1 Imprecise Dirichlet Model

For making probabilistic inferences from a set of observations about a discrete variable, Bayesian approaches [33] are commonly employed. Bayesian inferential methods tend to assume a *prior* distribution about p_{N+1} based on some parameters. Then, they take the *posterior* expectation of the parameters given the sample.

Specifically, Bayesian approaches usually assume a prior Dirichlet distribution for p_{N+1}. Such a distribution depends on a parameter $s > 0$ (real number) and a t-dimensional vector of non-negative real numbers $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_t)$ satisfying $\sum_{i=1}^{t} \alpha_i = 1$. The density function of the Dirichlet distribution takes the form:

$$f(p_{N+1}) = \frac{\Gamma(s)}{\prod_{i=1}^{t} \Gamma(s\alpha_i)} \prod_{i=1}^{t} p_{N+1}(x_i)^{s\alpha_i - 1}, \qquad (2.54)$$

Γ being the gamma function, which, for real numbers, is defined as follows:

$$\Gamma(z) = \int_0^\infty y^{z-1} e^{-y} dy, \qquad \forall z \in \mathbb{R}^+. \qquad (2.55)$$

In Bayesian inferential methods, the likelihood function of p_{N+1} given the sample $\mathcal{D}$, $L(p_{N+1} \mid \mathcal{D})$, is also considered:

$$L(p_{N+1} \mid \mathcal{D}) = \prod_{i=1}^{t} p_{N+1}(x_i)^{n_i}. \qquad (2.56)$$

[3] or, alternatively, a finite set of possible alternatives $\{x_1, x_2, \dots, x_t\}$.

Then, the prior Dirichlet Distribution is combined with the likelihood function via the Bayes's theorem to obtain the posterior expectation of p_{N+1} given the sample. For each x_i, the posterior expectation of $p_{N+1}(x_i)$ is given by:

$$p_{N+1}(x_i) = \frac{n(x_i) + s\alpha_i}{N+s}, \quad \forall i = 1, 2, \ldots, t. \tag{2.57}$$

The Imprecise Dirichlet Model (IDM), proposed by Walley [209], assumes a set of prior Dirichlet distributions, unlike the Bayesian approaches commented above, which assume a single prior distribution.

Specifically, the IDM assumes a fixed s and all possible values of the parameter vector α, i.e, all possible combinations of values $\alpha_i \in [0,1]$ verifying $\sum_{i=1}^{t} \alpha_i = 1$. Therefore, the IDM estimations for p_{N+1} are given by a set of probability intervals:

$$p_{N+1}(x_i) \in \left[\frac{n(x_i)}{N+s}, \frac{n(x_i)+s}{N+s}\right], \quad \forall i = 1, 2, \ldots, t. \tag{2.58}$$

Hence, we have the following set of IDM probability intervals:

$$\mathcal{I}_{IDM} = \left\{\left[\frac{n(x_i)}{N+s}, \frac{n(x_i)+s}{N+s}\right], \quad i = 1, 2, \ldots, t\right\}. \tag{2.59}$$

It can be deduced that, for each $A \subset \{x_1, x_2, \ldots, x_t\}$, the IDM predicts that the probability that the value of the next observation is in A belongs to the following interval:

$$P_{N+1}(A) \in \left[\frac{n(A)}{N+s}, \frac{n(A)+s}{N+s}\right]. \tag{2.60}$$

In this way, the IDM is also determined by the following coherent lower probability function:

$$\underline{P}_{IDM}(A) = \begin{cases} \frac{n(A)}{N+s} & \text{if } A \subset \{x_1, x_2, \ldots, x_t\} \\ 1 & \text{if } A = \{x_1, x_2, \ldots, x_t\} \end{cases} \tag{2.61}$$

Its dual coherent upper probability function is determined, for each $A \subseteq \{x_1, x_2, \ldots, x_t\}$, in the following way:

$$\overline{P}_{IDM}(A) = \begin{cases} \frac{n(A)+s}{N+s} & \text{if } A \subseteq \{x_1, x_2, \ldots, x_t\}, \quad A \neq \emptyset \\ 0 & \text{if } A = \emptyset \end{cases} \tag{2.62}$$

The choice of the parameter s is an essential issue in the inferences as it determines the imprecision degree [32]. Indeed, it is easy to observe that IDM probability intervals are wider as the s value is larger. Walley [209] does not give an absolute recommendation about the parameter s, but he suggests two values: $s = 1$ and $s = 2$.

2.3.1.1 *Inference principles with the IDM*

As argued in [32], inferences with the IDM satisfy the following principles, which were established as suitable for inference:

- *Symmetry principle* (SP): The prior probability assumed for p_{N+1} do not depend on the permutations of the values of X.
- *Likelihood principle* (LP): Inferences about the posterior probabilities given the observations depend on the sample only through the likelihood function, defined in Equation (2.56).
- *Representation invariance principle* (RIP) [209]: Coarsening or refinements of the set of possible values of X do not affect inferences about a certain subset $A \subseteq X$ because such inferences only depend on the sample size (N), the parameter s, and the number of observations in the sample whose value of X belongs to A $(n(A))$.

2.3.1.2 *IDM credal sets*

As shown by Abellán [2], the set of IDM probability intervals determined by Equation (2.59) is reachable and has the following credal set associated with it, composed of all probability distributions consistent with such intervals:

$$\mathcal{P}\left(\mathcal{I}_{IDM}\right) = \left\{ p \in \mathcal{P}(X) \mid p\left(x_i\right) \in \left[\frac{n(x_i)}{N+s}, \frac{n(x_i)+s}{N+s}\right], \quad \forall i = 1, 2, \dots, t \right\}. \tag{2.63}$$

We may note that the credal set defined in Equation (2.63) is also the one compatible with the IDM coherent lower probability function, determined via Equation (2.61).

Abellán [2] proved that the IDM credal set $\mathcal{P}\left(\mathcal{I}_{IDM}\right)$, defined in Equation (2.63), has the following properties:

- When $s = 0$, $\mathcal{P}\left(\mathcal{I}_{IDM}\right)$ has a single probability distribution p determined by relative frequencies, i.e, $p\left(x_i\right) = \frac{n(x_i)}{N}, \quad \forall i = 1, 2, \dots, t$. $\mathcal{P}\left(\mathcal{I}_{IDM}\right)$ is larger as the s value increases.
- $\mathcal{P}\left(\mathcal{I}_{IDM}\right)$ has t extreme points $e_1, e_2, \dots, e_t$, where $e_i = \left(e_i\left(x_1\right), e_i\left(x_2\right), \dots, e_i\left(x_t\right)\right)$, is determined in the following way:

$$e_i = \left(\frac{n(x_1)}{N+s}, \dots, \frac{n(x_i)+s}{N+s}, \dots, \frac{n(x_t)}{N+s}\right), \quad \forall i = 1, 2, \dots, t. \tag{2.64}$$

- The credal set $\mathcal{P}(\mathcal{I}_{IDM})$ is also representable by a belief function. Such a belief function coincides with the IDM coherent lower probability function defined in Equation (2.61). Its corresponding BPA (Möbius inverse), m_{IDM}, is given by:

$$\begin{aligned} &m_{IDM}(\{x_i\}) = \frac{n(x_i)}{N+s}, \quad \forall i = 1, 2, \ldots, t, \\ &m_{IDM}(A) = 0, \quad \forall A \subseteq \{x_1, x_2, \ldots, x_t\}, \mid 2 \leqslant |A| < t, \\ &m_{IDM}(\{x_1, x_2, \ldots, x_t\}) = \frac{s}{N+s}. \end{aligned} \tag{2.65}$$

  IDM credal sets are representable by belief functions and reachable probability intervals. However, as shown by Abellán [2], they are not the only credal sets that belong to both imprecise probability theories.

2.3.2 Non-Parametric Predictive Inference Model for multinomial data

The basis of the Non-Parametric Predictive Inference Model (NPI) [27] is Hill's assumption $A_{(N)}$ [113], defined as follows:

Definition 2.3.1 *Suppose that there are* N *observations* $y_1, y_2, \ldots, y_N$, *where* $y_i \in \mathbb{R} \quad \forall i = 1, 2, \ldots, N$. *Let us assume that there are no ties, that is,* $y_i \neq y_j \quad \forall i \neq j$. *Suppose that the observations are ordered so that* $y_1 < y_2 < \ldots < y_N$, *partitioning the real line into* $N+1$ *open intervals* $I_i = (y_i, y_{i+1})$, *for* $i = 0, 1, \ldots, N$, *where* $y_0 = -\infty$ *and* $y_{N+1} = \infty$. *Hill's assumption* $A_{(N)}$ *establishes that the next observation falls into any of these intervals with equal probability* $\frac{1}{N+1}$.

The previous assumption is useful when working with real-valued data. Nonetheless, in this thesis work, we focus on the Non-Parametric Predictive Inference Model for multinomial data (NPI-M) [58, 59]. Such a model employs an adaptation of Hill's assumption $A_{(N)}$ for multinomial data. This assumption is called circular $A_{(N)}$ [58].

Definition 2.3.2 *Suppose that we have* N *observations* $y_1, y_2, \ldots, y_N$ *that create* N *intervals on a circle, denoted by* $I_j = (y_j, y_{j+1}) \quad \forall j = 1, \ldots, N-1$ *and* $I_N = (y_N, y_1)$. *The circular assumption* $A_{(N)}$ *states that the probability that the next observation falls into any of these intervals is equal to* $\frac{1}{N}$.

The circular assumption $A_{(N)}$ is a post-data assumption related to exchangeability [58]. The same occurs with the original Hill's assumption.

In the NPI-M, a *probability wheel* is employed for representing the observed data, where each observation is represented via a line from the center of the wheel to its boundary, called *radial line*. The wheel is partitioned into N equally-sized slices. The NPI-M assumes that each possible value of the X variable can only be represented by a unique segment, where a segment is an area between two radial lines. Consequently, lines representing the same value must be positioned next to each other on the wheel. It has to be decided which value of X represents each slice. For this purpose, the following criteria are used:

- When two lines that represent the same value of X border to a slice, such a value must be assigned to that slice.
- If a slice is bordered by two lines that represent distinct values, then any of these two values or anyone not observed can be assigned to that slice.
- Also, more than one value can be assigned to a slice.

Suppose that we want to make inference about the probability that the value of the next observation belongs to a given subset $A \subseteq \{x_1, x_2, \dots, x_t\}$. The idea of the NPI-M for this inference is the following one [59]: an arrow, fixed at the center of the wheel, spins around the wheel. According to the circular assumption $\mathcal{A}_{(N)}$, the arrow has the same probability $\frac{1}{N}$ of stopping in each slice. Thus, for a given configuration of the probability wheel, the NPI-M predicts that the probability that the value of the next observation is in A is equal to the proportion of slices assigned to a value belonging to A in that configuration of the probability wheel. In this way, the NPI-M predicts a lower probability and an upper probability for A, determined, respectively, by the minimum and maximum proportion of slices that can be assigned to a value in A, among all possible configurations of the probability wheel.

Let t_{obs} be the number of values of X that have been observed and t_{unobs} the number of unobserved values of X:

$$\begin{aligned} t_{obs} &= |\{x_i \mid n(x_i) > 0, \quad i = 1, 2, \dots, t\}|, \\ t_{unobs} &= |\{x_i \mid n(x_i) = 0, \quad i = 1, 2, \dots, t\}|. \end{aligned}$$

Clearly, it holds that $t = t_{obs} + t_{unobs}$. Let t^A_{obs} and t^A_{unobs} denote, respectively, the number of observed and unobserved values in A. Let $n(A)$ be the number of observations in A:

$$\begin{aligned} n(A) &= \sum_{x_i \in A} n(x_i), \\ t^A_{obs} &= |\{x_i \in A \mid n(x_i) > 0\}|, \\ t^A_{unobs} &= |\{x_i \in A \mid n(x_i) = 0\}|. \end{aligned}$$

In [59], the following theorem was proven, which indicates how the NPI-M lower and upper probabilities for a given subset of possible values of X are obtained:

Theorem 2.3.1 *For each $A \subseteq \{x_1, x_2, \ldots, x_t\}$, the NPI-M lower and upper probabilities, based on the probability wheel described above from the N observations and the circular assumption $\mathcal{A}_{(N)}$, are determined by:*

$$\begin{aligned}\underline{P}_{NPI}(A) &= \frac{n(A) - \min\left(t - |\overline{A}|, t^{A}_{obs}\right)}{N},\\ \overline{P}_{NPI}(A) &= \frac{n(A) + \min\left(|A|, t_{obs} - t^{A}_{obs}\right)}{N},\end{aligned} \tag{2.66}$$

where $\overline{A}$ denotes the complement of A.

The idea of the NPI-M is illustrated in Example 2.3.1 [59]:

Example 2.3.1 *Suppose that we have a discrete variable called Color whose set of possible values is $\{\mathrm{Blue}(B), \mathrm{Red}(R), \mathrm{Yellow}(Y), \mathrm{Green}(G), \mathrm{White}(W), \mathrm{Orange}(O)\}$.*

Let us assume that there are $N = 9$ observations about Color, with the following observed frequencies for each value:

$$n(B) = 3, \quad n(R) = 1, \quad n(Y) = 2, \quad n(G) = 3, \quad n(W) = n(O) = 0.$$

Suppose that we want to compute the NPI-M lower probability of $\{B\}$ and $\{B, R\}$. For this purpose, we aim to find a configuration of the probability wheel that assigns as least slices as possible to B and R. In this case, we can assign to B only those slices bordered by two lines representing B, and we do not need to assign to R any slice. Thereby, Figure 2.2 shows a configuration of the probability wheel suitable for the NPI-M lower probability of $\{B\}$ and $\{B, R\}$, where the color drawn in each slice corresponds to the color assigned to that slice. In that configuration, only two slices are assigned to B and none to R.

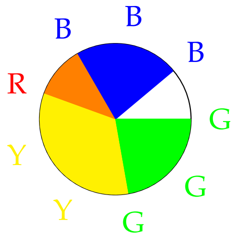

Figure 2.2: First configuration of the probability wheel of Example 2.3.1.

According to Theorem 2.3.1, it is satisfied that

$$\underline{P}_{NPI}(\{B\}) = \frac{2}{9} = \underline{P}_{NPI}(\{B, R\}).$$

These lower probabilities are consistent with the configuration of the probability wheel illustrated in Figure 2.2.

Let us assume now that it is needed to compute the upper probability of $\{B\}$ *and* $\{B, R\}$. *In this situation, we try to find a configuration of the probability wheel that assigns slices bordered by a line representing* B *to* B *and slices bordered by a line associated with* R *to* R. *Furthermore, it is possible to separate the lines corresponding to* R *and* B *by lines representing another color in order to assign as many slices as possible to* B *and* R. *A configuration that verifies the previous conditions can be seen in Figure 2.3. From that configuration, it can be deduced that the upper probability of* $\{B\}$ *is equal to* $\frac{4}{9}$ *and the upper probability of* $\{B, R\}$ *is equal to* $\frac{6}{9}$. *Indeed, according to Theorem 2.3.1, it holds that:*

$$\overline{P}_{NPI}(\{B\}) = \frac{4}{9}, \quad \overline{P}_{NPI}(\{B, R\}) = \frac{6}{9}.$$

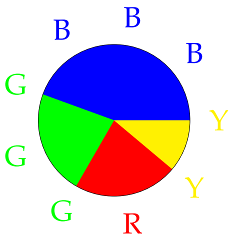

Figure 2.3: Second configuration of the probability wheel of Example 2.3.1.

2.3.2.1 *Properties of the NPI-M lower and upper probabilities*

The following essential properties of the NPI-M lower and upper probabilities were demonstrated in [59]:

1. $\overline{P}_{NPI}(A) = 1 - \underline{P}_{NPI}(\overline{A}), \quad \forall A \subseteq \{x_1, x_2, \ldots, x_t\}$.

2. Probabilities determined by relative frequencies are always consistent with the NPI-M lower and upper probabilities:

$$\underline{P}_{NPI}(A) \leqslant \frac{n(A)}{N} \leqslant \overline{P}_{NPI}(A), \quad \forall A \subseteq \{x_1, x_2, \ldots, x_t\}.$$

3. The NPI-M lower and upper probabilities, determined via Theorem 2.3.1, are, respectively, coherent lower and upper probability functions. Furthermore, due to the first property, they are dual or conjugate.

4. As the number of observations diverges to infinity, the NPI-M lower probability converges to the NPI-M upper probability.

2.3.3 Comparison with the IDM

The IDM assumes a set of prior Dirichlet distributions about the data through a parameter. In contrast, the NPI-M does not make previous assumptions about the data. The only assumption made by the NPI-M is the circular $\mathcal{A}_{(N)}$, which is a post-data assumption. NPI-M is a non-parametric approach.

As pointed in [59], the NPI-M do not satisfy the RIP, unlike the IDM. Even though the RIP was established as a crucial principle for inference by Walley [209], in [59], it was claimed that the fact that the NPI-M violates the RIP is not a shortcoming. What is more, since the NPI-M does not satisfy the RIP, inferences made with the NPI-M might produce intuitively more coherent results than inferences with the IDM.

The differences between the results of inferences about the next observation with the IDM and with the NPI-M were analyzed in [59]. The most relevant conclusions from such an analysis can be summarized in the following points:

- For inferences on a singleton $\{x_i\}$, with $i \in \{1, 2, \dots, t\}$, the NPI-M lower probability is greater than 0 if, and only if, there are at least two observations for which $X = x_i$. In contrast, if the x_i value has been observed only once, then the IDM lower probability is strictly greater than 0. It can be stated that, in that sense, the inferences made with the NPI-M are more conservative than the inferences made with the IDM.

- The number of non-observed values does not influence the IDM lower and upper probabilities. Moreover, the computation of the IDM lower and upper probabilities does not depend on the number of non-observed values included in the inference subset. It does not happen with the NPI-M since the number of observed and unobserved values and how many are included in the inference subset influence the determination of the NPI-M lower and upper probabilities.

- When the inference subset just contains non-observed values, the IDM upper probability only depends on the parameter s and the total number of observations N, while the NPI-M upper probability is influenced by the number of observed and unobserved values and the cardinality of the inference subset.

- In IDM inferences, the imprecision (the difference between the upper and lower probabilities) does not change when the cardinality of the inference subset varies. In contrast, with the NPI-M, the imprecision in an inference increases as the inference subset is larger. Intuitively, it makes sense that the imprecision is higher as the cardinality of the inference subset increases.

To sum up, it can be stated that NPI-M inferences may provide more intuitive results than IDM inferences.

2.3.4 Approximate Non-Parametric Predictive Inference Model

We may note that, for singletons $\{x_i\}$, with $i \in \{1, 2, \ldots, t\}$, the NPI-M lower and upper probabilities are given by:

$$\underline{P}_{NPI}\left(\{x_i\}\right) = \max\left(\frac{n(x_i) - 1}{N}, 0\right), \quad \overline{P}_{NPI}\left(\{x_i\}\right) = \min\left(\frac{n(x_i) + 1}{N}, 1\right). \tag{2.67}$$

Hence, we have the following set of NPI-M probability intervals for singletons:

$$\mathcal{I}_{NPI} = \left\{\left[\max\left(\frac{n(x_i) - 1}{N}, 0\right), \min\left(\frac{n(x_i) + 1}{N}, 1\right)\right], \quad i = 1, 2, \ldots, t\right\}. \tag{2.68}$$

As demonstrated in [5], the set of probability intervals given in Equation (2.68) is reachable and has associated with it the following credal set:

$$\begin{aligned} &\mathcal{P}\left(\mathcal{I}_{NPI}\right) = \{p \in \mathcal{P}(X) \mid p\left(x_i\right) \in \\ &\left[\max\left(\frac{n(x_i) - 1}{N}, 0\right), \min\left(\frac{n(x_i) + 1}{N}, 1\right)\right], \quad \forall i = 1, 2, \ldots, t\right\}. \end{aligned} \tag{2.69}$$

Moreover, the following result was proven in [5]:

Proposition 2.3.1 *The natural extension of the set of NPI-M probability intervals for singletons, defined in Equation (2.68), coincides with the NPI-M coherent lower probability function, determined by Theorem 2.3.1.*

In consequence, the NPI-M lower and upper probabilities of each $A \subseteq \{x_1, x_2, \ldots, x_t\}$ can be obtained from the NPI-M lower and upper probabilities for singletons. However, there may be some probability distributions in the credal set corresponding to the NPI-M probability intervals for singletons, defined in Equation (2.69), that do not satisfy the constraints imposed by the NPI-M probability wheel representation of the data. In fact, the set of probability distributions compatible with the NPI-M is not convex. This was shown with an example in [5].

By considering the set of probability distributions belonging to the credal set associated with the NPI-M probability intervals for singletons, an approximate model, called Approximate Non-Parametric Predictive Inference Model for multinomial data (A-NPI-M), is obtained [5]. It corresponds to the convex hull of the set of probability distributions compatible with the NPI-M. Therefore, the A-NPI-M simplifies the exact model as it avoids some constraints imposed by the NPI-M probability wheel representation of the data.

3 QUANTIFICATION OF THE UNCERTAINTY-BASED INFORMATION WITH IMPRECISE PROBABILITIES

3.1 Introduction

When mathematical theories based on imprecise probabilities arise, new tools for quantifying the uncertainty-based information in such theories are needed. These tools are often called *uncertainty measures*. The study of uncertainty measures in imprecise probability theories has its origin in the uncertainty measures in classical information theories.

On the one hand, in classical possibility theory, the *Hartley measure* [110] was established as suitable for quantifying uncertainty-based information. Such a measure consists of a function that depends on the cardinality of the subset to which the real alternative belongs. The type of uncertainty captured by the Hartley measure is known as *non-specificity*.

On the other hand, the uncertainty in classical probability theory is well-measured via the Shannon entropy [189], which quantifies the uncertainty-based information involved in a probability distribution. The type of uncertainty measured by the Shannon entropy is called *conflict* or *discord*.

In some imprecise probability theories, uncertainty measures have been developed, where it has been assumed that both conflict and non-specificity coexist. Thereby, both the Hartley measure and the Shannon entropy must be properly extended to such theories. The study of uncertainty measures in Evidence theory (ET) is the basis of uncertainty measures in more general imprecise probability theories. In fact, the uncertainty measures in imprecise probabilities proposed so far consist of extensions of uncertainty measures initially developed in ET.

This chapter is structured as follows: Section 3.2 details the study of uncertainty in classical theories (possibility and probability theories). The research carried out so far concerning uncertainty measures in Evidence theory is described in Section 3.3. In Section 3.4, we present the main uncertainty measures proposed so far in more general imprecise probability theories.

Within this chapter, it is assumed that $X = \{x_1, x_2, \dots, x_t\}$ is a finite set[1], with $|X| = t$. $\wp(X)$ will denote the power set of X and $\mathcal{P}(X)$ the set of all probability distributions on X.

3.2 Uncertainty in classical information theories

3.2.1 Uncertainty in possibility theory

Definition 3.2.1 *[110] The function given by:*

$$H(A) = \log_2(|A|), \quad \forall A \in \wp(X) \tag{3.1}$$

is called Hartley measure.

The type of uncertainty captured by H is usually known as *non-specificity*. The Hartley measure is the most suitable uncertainty measure in classical possibility theory. Indeed, as demonstrated by Hartley [110], it is the only function defined in terms of the cardinality of a subset that satisfies the three following desirable properties:

1. **Normalization**: If $A \subset X$ verifies that $|A| = 2$, then $H(A) = 1$.
2. **Monotonicity**: $H(B) \leqslant H(A) \quad \forall A, B \in \wp(X)$ such that $B \subseteq A$.
3. **Additivity**: Let X and Y be two finite sets, $A \subseteq X$, and $B \subseteq Y$. Then, $H(A \times B) = H(A) + H(B)$.

Also, it can be easily deduced that the maximum value of the Hartley measure is equal to $\log_2(|X|)$. It is reached when $A = X$.

3.2.2 Uncertainty in probability theory

Definition 3.2.2 *[189] Let p be a probability distribution on X. The function determined by:*

$$S(p) = -\sum_{i=1}^{t} p\left(x_i\right) \log_2(p(x_i)) \tag{3.2}$$

is known as the Shannon entropy on p.

[1] Or a discrete variable that takes values in $\{x_1, x_2, \dots, x_t\}$.

The Shannon entropy is the well-established uncertainty measure in classical probability theory. It captures a type of uncertainty often called *conflict* or *discord*. This type of uncertainty is different from the one quantified by the Hartley measure.

As proved in [189], the Shannon entropy is derived by using the three following axioms:

1. **Continuity**: Little variations in $p(x_i)$, for a certain $i \in \{1, 2, \dots, t\}$, must lead to small changes in $S(p)$.
2. **Monotonicity**: For uniform probability distributions, S must be an increasing function of the number of alternatives.

   Formally, let $X = \{x_1, x_2, \dots, x_t\}$ and $Y = \{y_1, y_2, \dots, y_{t'}\}$ be two finite sets such that $t \leqslant t'$. Let p_u^X and p_u^Y be the uniform probability distributions on X and Y, respectively, that is,

   $$p_u^X(x_i) = \frac{1}{t}, \quad \forall i = 1, 2, \dots, t,$$

   $$p_u^Y(y_j) = \frac{1}{t'}, \quad \forall j = 1, 2, \dots, t'$$

   Then, it must hold that $S(p_u^X) \leqslant S(p_u^Y)$.
3. **Additivity**: Let X and Y be two finite sets and p a probability distribution on the product space $X \times Y$. Let $p^{\downarrow X}$ and $p^{\downarrow Y}$ denote the marginal probability distributions of p on X and Y, respectively:

   $$p^{\downarrow X}(x) = \sum_{y \in Y} p(x, y), \quad \forall x \in X,$$

   $$p^{\downarrow Y}(y) = \sum_{x \in X} p(x, y), \quad \forall y \in Y.$$

   Suppose that the marginal probability distributions are independent. Then, S must verify that

   $$S(p) = S(p^{\downarrow X}) + S(p^{\downarrow Y}).$$

The Shannon entropy also satisfies the following desirable properties:

- S does not depend on the arrangement of the elements of X, only on their probabilities.
- $S(p) \geqslant 0$, with equality if, and only if, $p(x_i) = 1$ for some $i \in \{1, 2, \dots, t\}$.
- The maximum value of S, which is equal to $\log_2(|X|)$, is attained when p is the uniform probability distribution on X, i.e, $p(x_i) = \frac{1}{t}, \quad \forall i = 1, 2, \dots, t$.

3.3 Uncertainty measures in Evidence theory

According to Yager [222], in Evidence theory (ET), both conflict and non-specificity coexist. Conflict appears when the uncertainty-based information resides in subsets with empty intersection, while non-specificity arises when the uncertainty-based information focuses on subsets with cardinality greater than one. Therefore, both the Hartley measure (the well-established non-specificity measure in classical possibility theory) and the Shannon entropy (the suitable conflict measure in classical probability theory) must be properly generalized to ET.

3.3.1 Essential mathematical properties and behavioral requirements for uncertainty measures in Evidence theory

Klir and Wierman [127] carried out a study about the crucial mathematical properties that have to be satisfied by total uncertainty measures in ET. According to such a study, a total uncertainty measure in ET, UM, that jointly quantifies conflict and non-specificity should satisfy the following five fundamental properties:

1. **Probabilistic consistency**: If m is a BPA on X such that all its focal elements are singletons, then UM must coincide with the Shannon entropy, that is:
$$UM(m) = -\sum_{i=1}^{t} m\left(\{x_i\}\right)\log_2\left(m\left(\{x_i\}\right)\right).$$

2. **Set Consistency**: Let m be a BPA on X. Suppose that there exists a subset $A \subseteq X$ such that $m(A) = 1$. Then, UM must collapse to the Hartley measure:
$$UM(m) = \log_2(|A|).$$

3. **Range**: The range of UM has to be equal to $[0, \log_2(t)]$.

4. **Subadditivity**: Let X and Y be two finite sets and m a BPA on the product space $X \times Y$. Let $m^{\downarrow X}$ and $m^{\downarrow Y}$ denote the respective marginal BPAs of m on X and Y, determined via Equations (2.36) and (2.37), respectively. Then, UM must verify the following inequality:
$$UM(m) \leqslant UM(m^{\downarrow X}) + UM(m^{\downarrow Y}).$$

   The idea of this property is that, when a BPA defined on a joint space is decomposed, the uncertainty-based information must not be increased.

5. **Additivity**: Let m be a BPA defined on a product space $X \times Y$, X and Y being two finite sets. Let $m^{\downarrow X}$ and $m^{\downarrow Y}$ be the respective marginal BPAs of m on X and Y. Suppose that the marginal BPAs are non-interactive. Then, it must hold that

$$UM(m) = UM(m^{\downarrow X}) + UM(m^{\downarrow Y}).$$

This means that, when a BPA defined on a product space such that the marginal BPAs are independent is decomposed, the uncertainty-based information has to be preserved.

In some cases, depending on the form of the uncertainty measure, it makes more sense to consider the submultiplicativity and multiplicativity properties than subadditivity and additivity. Such properties are defined in the following way taking into account the definitions of additivity and subadditivity:

- **Submultiplicativity**: Let X and Y be two finite sets and m a BPA on the product space $X \times Y$. Let $m^{\downarrow X}$ and $m^{\downarrow Y}$ denote the respective marginal BPAs of m on X and Y. Then, UM must verify that:

$$UM(m) \leqslant UM(m^{\downarrow X}) \times UM(m^{\downarrow Y}).$$

- **Multiplicativity**: Let m be a BPA defined on a product space $X \times Y$, where X and Y are two finite sets. Let $m^{\downarrow X}$ and $m^{\downarrow Y}$ denote the marginal BPAs of m on X and Y, respectively. Suppose that the marginal BPAs are non-interactive. Then, it must hold that

$$UM(m) = UM(m^{\downarrow X}) \times UM(m^{\downarrow Y}).$$

Submultiplicativity and multiplicativity are essentially equivalent to subadditivity and additivity [224].

We may note that the properties described above are based on the essential mathematical properties satisfied by the Hartey measure and the Shannon entropy in classical information theories. Since ET is more general than possibility and probability theories, in ET, situations that never happen in classical theories can arise.

The study carried out by Klir and Wierman was extended by Abellán and Masegosa [21] by taking the following point into consideration: in probability theory, a probability distribution can never be contained in another probability distribution. In contrast, in ET, the uncertainty-based information involved in a BPA can contain the uncertainty-based information involved in another BPA. This issue must be taken into account by every total uncertainty measure in ET. Thus, the following property is also considered as crucial [21]:

- **Monotonicity**: Let m_1 and m_2 be two BPAs on X. Let $\mathcal{P}_{m_i}$ denote the credal set consistent with the BPA m_i, for $i = 1, 2$. Suppose that $\mathcal{P}_{m_1} \subseteq \mathcal{P}_{m_2}$. Then, it must hold that:

$$UM(m_1) \leqslant UM(m_2).$$

An uncertainty measure in ET not only has to satisfy the crucial mathematical properties but its behavior in different situations must be desirable. In this way, Abellán and Masegosa [21] also claimed that every total uncertainty measure in ET, UM, must satisfy the following behavioral requirements:

1. **Computational complexity**: The calculation of UM must not be too complex.

2. **Coherent disaggregation**: UM must not conceal the two types of uncertainty that appear in ET: conflict and non-specificity. Consequently, it has to be possible to decompose UM into two measures that coherently quantify conflict and non-specificity, respectively.

3. **Sensitivity to changes in the BPA**: UM has to be sensitive to changes in the BPA. However, it must be considered that, sometimes, an increase of conflict might produce a decrease of non-specificity and vice-versa. Hence, there can be similar values of UM with different values of conflict and non-specificity. For this reason, UM has not to be sensitive to changes in the BPA directly, it can also be sensitive through its parts of conflict and non-specificity.

In some situations, it is more appropriate to mathematically quantify the available information through more general theories than ET [127]. In these cases, the *principle of uncertainty invariance* must be taken into account, which establishes that when a representation of uncertainty in a mathematical theory is transformed into its counterpart in another theory, the amount of information must be preserved. Thereby, every uncertainty measure in ET, UM, must satisfy the following behavioral requirement:

- **Generalization**: The extension of UM to more general theories than ET must be possible.

3.3.2 Main approaches for quantifying uncertainty in Evidence theory

The Hartley measure was extended to ET by Dubois and Prade [83]. Given a BPA m on X, the Generalized Hartley Measure is defined in the following way:

$$GH(m) = \sum_{A \in \wp(X)} m(A) \log_2(|A|). \tag{3.3}$$

When m is a probability distribution (all its focal elements are singletons), GH reaches its minimum value, which is equal to 0. The maximum value of GH is equal to $\log_2(|X|)$. It is attained when $m(X) = 1$. As shown in [84], GH is a suitable non-specificity measure in ET as it satisfies the essential mathematical properties. In addition, it is possible to extend it to more general imprecise probability theories [12].

There were several attempts to generalize the Shannon entropy to ET. One of the most remarkable was the Dissonance measure of Yager [222]. It is defined, for a given BPA m on X, as follows:

$$E(m) = - \sum_{A \in \wp(X)} m(A) \log_2 Pl_m(A), \tag{3.4}$$

where Pl_m is the plausibility function corresponding to m.

However, none of the proposed extensions of the Shannon entropy to ET satisfies the essential subadditivity property [127]. It would have been acceptable if this requirement had been satisfied by the total uncertainty measure resulting from summing GH and the generalized Shannon entropy, but it did not happen for any of the proposed extensions of the Shannon entropy.

At the middle of 90's, these unsuccessful attempts of generalizing the Shannon entropy were replaced by a total uncertainty measure that captures both conflict and non-specificity. That measure, presented by Harmanec and Klir [109], consists of the maximum entropy on the credal set compatible with a BPA. Formally, for a given BPA m on X, the maximum entropy is determined as follows:

$$S^* \left(\mathcal{P}(Bel_m)\right) = \max_{p \in \mathcal{P}(Bel_m)} S(p), \tag{3.5}$$

Bel_m being the belief function associated with m and $\mathcal{P}(Bel_m)$ the credal set consistent with Bel_m, computed via Equation (2.32).

As pointed out by Abellán and Masegosa [21], S^* satisfies all required mathematical properties for uncertainty measures in ET. Furthermore, this measure can be easily extended to more general imprecise probability theories [1].

When the maximum entropy was proposed, it did not separate conflict and non-specificity. As we have shown, this is a fundamental behavioral requirement for total uncertainty measures in ET. Smith [192] proposed the following disaggregation of $S^*\left(\mathcal{P}(Bel_m)\right)$:

$$S^*\left(\mathcal{P}(Bel_m)\right) = GH + \left(S^*\left(\mathcal{P}(Bel_m)\right) - GH\right). \tag{3.6}$$

The first term of Equation (3.6) quantifies non-specificity whereas the second term captures conflict. It always holds that $S^*\left(\mathcal{P}(Bel_m)\right) - GH \geqslant 0$ and, thus, it is meaningful to consider $S^*\left(\mathcal{P}(Bel_m)\right) - GH$ as the generalization of the Shannon entropy to ET [192].

This decomposition of S^* is not unique. Indeed, Abellán and Moral [15] proposed the following disagreggation of S^*[2]:

$$S^*\left(\mathcal{P}(Bel_m)\right) = S_*\left(\mathcal{P}(Bel_m)\right) + \left(S^*\left(\mathcal{P}(Bel_m)\right) - S_*\left(\mathcal{P}(Bel_m)\right)\right), \tag{3.7}$$

where $S_*\left(\mathcal{P}(Bel_m)\right)$ is the minimum entropy on $\mathcal{P}(Bel_m)$:

$$S_*\left(\mathcal{P}(Bel_m)\right) = min_{p \in \mathcal{P}(Bel_m)} S(p). \tag{3.8}$$

In the expression given by Equation (3.7), the first term captures conflict while the second term quantifies non-specificity. Abellán and Moral [15] demonstrated that $\left(S^*\left(\mathcal{P}(Bel_m)\right) - S_*\left(\mathcal{P}(Bel_m)\right)\right)$ is a suitable non-specificity measure. Indeed, it does not satisfy the subadditivity property, but it does not matter since the total uncertainty measure S^* verifies this requirement.

Some works in the literature, such as [123, 211], criticised that the maximum entropy is not sensitive to changes in the BPA. Indeed, it is not directly sensitive. Nonetheless, Abellán and Masegosa [21] showed that the maximum entropy is sensitive to changes in a BPA via its parts of conflict and non-specificity. As commented before, it makes sense because an increase (decrease) of conflict may lead to a decrease (increase) of non-specificity even though the total uncertainty value does not vary.

Despite the previous points, the algorithms proposed so far for the computation of the maximum entropy (see [109, 118, 145, 154]) are notably complex. This supposes a drawback for using such a measure in practical applications. For this reason, many alternative measures to the maximum entropy have been developed during the last years.

- Jousselme et. al [123] proposed a total uncertainty measure that consists of the Shannon entropy of the pignistic transformation of a BPA. That

[2] The decomposition was proposed for the generalization of the maximum entropy for credal sets, but the decomposition is also useful in ET.

measure is called *ambiguity measure* (AM). Formally, for a given BPA m on X, the ambiguity measure is defined as follows:

$$AM(m) = \sum_{i=1}^{t} BetP_m\left(\{x_i\}\right) \log_2\left(BetP_m\left(\{x_i\}\right)\right), \tag{3.9}$$

where $BetP_m$ is the pignistic transformation of the BPA m, computed by means of Equation (2.31).

Klir and Lewis [129] found some drawbacks on this ambiguity measure. In order to solve such shortcomings, Shahpari and Seyedin [188] presented a modified ambiguity measure (MAM). On 1-D space, MAM and AM coincide. Nonetheless, for a 2-D space, Shahpari and Seyedin utilized a different definition for the pignistic transformation and they did not substantially justified it.

Formally, let $X = \{x_1, x_2, \ldots, x_t\}$ and $Y = \{y_1, y_2, \ldots, y_{t'}\}$ be two finite sets and m a BPA defined on the product space $X \times Y$. On $X \times Y$, AM coincides with MAM. Now, for the marginal BPA of m on X, Shahpari and Seyedin [188] employed the following probability distribution:

$$MBet_{mX}\left(\{x_i\}\right) = \sum_{A \subseteq X \times Y | x_i \in A^{\downarrow X}} \frac{m(A)\#(x_i \in A)}{|A|}, \quad \forall i = 1, 2, \ldots, t, \tag{3.10}$$

$A^{\downarrow X}$ being the projection of A on X and $\#(x_i \in A)$ the number of occurrences of x_i in A, $\forall i \in \{1, 2, \ldots, t\}$, $A \subseteq X \times Y$.

Analogously, Shahpari and Seyedin [188] used the following probability distribution for the marginal BPA of m on Y:

$$MBet_{mY}\left(\{y_j\}\right) = \sum_{A \subseteq X \times Y | y_j \in A^{\downarrow Y}} \frac{m(A)\#(y_j \in A)}{|A|}, \quad \forall j = 1, 2, \ldots, t', \tag{3.11}$$

where $A^{\downarrow Y}$ is the projection of A on Y and $\#(y_j \in A)$ the number of occurrences of y_j in A, $\forall j \in \{1, 2, \ldots, t'\}$, $A \subseteq X \times Y$.

In this way, on the marginal BPAs of m on X and Y, Shahpari and Seyedin [188] considered the Shannon entropy on the probability distributions on X and Y determined by Equations (3.10) and (3.11), respectively.

Abellán and Bossé [7] pointed out that both AM and MAM satisfy probabilistic consistency, set consistency, and range; both AM and MAM violate additivity and monotonicity; AM does not verify subadditivity; MAM satisfies the subadditivity property in a controversial way because of the definition of MAM for marginal BPAs.

- One of the most frequently used uncertainty measures proposed in the last years is the *Deng entropy* [77]. Given a BPA m on X, the Deng entropy is defined in the following way:

$$\begin{gathered} E_{Deng}(m) = -\sum_{A \in \wp(X)} m(A) \log_2 \left(\frac{m(A)}{2^{|A|}-1} \right) = \\ \sum_{A \in \wp(X)} m(A) \log_2 \left(2^{|A|} - 1 \right) - \sum_{A \in \wp(X)} m(A) \log_2 (m(A)). \end{gathered} \tag{3.12}$$

In Equation (3.12), the first term indicates non-specificity whereas the second term quantifies conflict. The idea of this measure is that the uncertainty must be considerably increased as there are more alternatives.

The Deng entropy has been commonly employed in the literature [69, 125, 237]. However, Abellán [4] demonstrated that this measure violates most of the essential mathematical properties for total uncertainty measures in ET. Actually, among the crucial mathematical properties for total uncertainty measures in ET, detailed in Section 3.3.1, only the probabilistic consistency is satisfied by the Deng entropy. Moreover, the behavior of this uncertainty measure in some situations is questionable. For example, when all focal elements share an element, the conflict part of the Deng entropy might not be equal to 0. This is a shortcoming because, in these cases, there is no conflict [4]. Also, the extension of the Deng entropy to more general theories than ET is still an open question.

In [236], a modification of the Deng entropy, known as Zhou entropy, was proposed. It is defined, for a given BPA m on X, as follows:

$$\begin{aligned} E_{Zhou}(m) &= -\sum_{A \in \wp(X)} m(A) \log_2 \left(\frac{m(A)}{2^{|A|}-1} \exp\left(\frac{|A|-1}{|X|} \right) \right) \\ &= \sum_{A \in \wp(X)} m(A) \log_2 (2^{|A|} - 1) - \\ &\quad \sum_{A \in \wp(X)} m(A) \log_2 \left(\exp\left(\frac{|A|-1}{|X|} \right) \right) - \sum_{A \in \wp(X)} m(A) \log_2 m(A). \end{aligned} \tag{3.13}$$

It could be considered that the first two terms in Equation (3.13) quantify the non-specificity part in a BPA since both of them are equal to 0 when m is a probability distribution. The third one might measure the conflict part, which is the same as in the original Deng entropy. It can be observed that E_{Zhou} is also based on the idea of the Deng entropy as it gives a higher total uncertainty value when the number of alternatives increases. However, due to the second term, with the modification, this increase is better controlled.

Afterwards, Cui, Liu, Zhang, and Kang [69] proposed a new version of the Deng entropy that takes the intersections between the focal elements into consideration. It is defined in the following way:

$$\begin{aligned} E_{Cui}(m) &= -\sum_{A\in\wp(X)} m(A)\log_2\left[\left(\frac{m(A)}{2^{|A|}-1}\right)\exp\left(\sum_{B\neq A\wedge m(B)>0}\frac{|A\cap B|}{2^{|X|-1}}\right)\right] \\ &= \sum_{A\in\wp(X)} m(A)\log_2\left(2^{|A|}-1\right) \\ &- \sum_{A\in\wp(X)} m(A)\log_2\left[m(A)\times\exp\left(\sum_{B\neq A\wedge m(B)>0}\frac{|A\cap B|}{2^{|X|-1}}\right)\right]. \end{aligned} \tag{3.14}$$

- Jirousek and Shenoy [122] presented a total uncertainty measure based on the *plausibility transformation* of a BPA [57, 207]. Given a BPA m on X, the mentioned transformation is defined as follows:

$$Pt_m(\{x_i\}) = \frac{Pl_m(\{x_i\})}{\sum_{j=1}^{t} Pl_m\left(\{x_j\}\right)}, \quad \forall i = 1,2,\dots,t. \tag{3.15}$$

For a given BPA m on X, the uncertainty measure proposed by Jirousek and Shenoy is determined by the sum of $GH(m)$ and the Shannon entropy of the plausibility transformation of m [122]:

$$E_{JS}(m) = GH(m) - \sum_{i=1}^{t} Pt_m(\{x_i\})\log_2(Pt_m(\{x_i\})). \tag{3.16}$$

The first term of Equation (3.16) is associated with non-specificity, whereas the second term captures conflict.

As demonstrated in [122], E_{JS} satisfies probabilistic consistency, additivity, and monotonicity, but it violates set consistency and subadditivity. The range of E_{JS} is equal to $[0, 2\log_2(t)]$ and, thus, E_{JS} does not satisfy the range property [122].

The computation of this E_{JS} is fast. It separates conflict and non-specificity. However, we can observe that the conflict value of E_{JS} may not be equal to 0 when all focal elements are not disjunct. As explained before, it is an undesirable behavior. In addition, the extension of E_{JS} to more general theories than ET is still an open problem.

- An uncertainty measure that also uses the plausibility transformation was proposed in [170]. It is defined in the following way:

$$E_{PQ}(m) = -\sum_{A\subseteq X} m(A)\log_2\left(Pm(A)\right) + GH(m), \qquad (3.17)$$

  where $Pm(A) = \sum_{x_i \in A} Pt_m\left(\{x_i\}\right), \quad \forall A \subseteq X$. The first term captures conflict, and the second one corresponds to non-specificity.

  E_{PQ} satisfies probabilistic consistency, monotonicity, and additivity. Nevertheless, it does not verify neither range nor subadditivity [170]. When $m(A) = 1$ for some $A \subseteq X$, we may note that $E_{PQ}(m) = \log_2(Pm(A)) + \log_2(|A|)$. Consequently, E_{PQ} does not satisfy set consistency.

  It can be observed that E_{PQ} separates conflict and non-specificity. Nonetheless, we may note that, when all focal elements have an element in common, the conflict part of E_{PQ} might be greater than 0, which is not desirable. Furthermore, the extension of E_{PQ} to more general theories than ET is still an open problem.

- Zhao et. al [235] proposed a total uncertainty measure that combines the Deng entropy with the belief intervals for singletons. It is defined, for a given BPA m on X, as follows:

$$\begin{aligned} E_{inter}(m) &= -\sum_{i=1}^{t} \frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2} \\ &\times \log_2\left[\frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2} \times \exp(-(Pl_m(\{x_i\}) - Bel_m(\{x_i\})\right] \\ &- \sum_{A\subseteq X||A|\geqslant 2} m(A) \times \log_2\left[\frac{m(A)}{2^{|A|}-1}\exp(-(Pl_m(A) - Bel_m(A)))\right]. \end{aligned} \qquad (3.18)$$

  In the previous expression, the first term corresponds to conflict, while the second one indicates non-specificity.

  This total uncertainty measure verifies probabilistic consistency, but it violates set consistency, range, subadditivity, and additivity [235]. Although the monotonicity property was not formally proved in [235], in that work, it was illustrated via numerical examples that $E_{inter}(m)$ might satisfy this requirement.

  $E_{inter}(m)$ separates conflict and non-specificity. Nevertheless, when all focal elements of m share an element, the conflict part of $E_{inter}(m)$ might not be equal to 0, which is an undesirable behavior. In addition, it must be remarked that the extension of $E_{inter}(m)$ to more general theories than ET is still an open problem.

Table 3.1: Summary of the essential mathematical properties verified by the uncertainty measures proposed so far in ET. UM = uncertainty measure, Prob con = Probabilistic consistency, Set con = Set consistency, Subadd = subadditivity, and Mon = monotonicity.

UM	Prob con	Set con	Range	Subadd	Add	Mon
S^*	Yes	Yes	Yes	Yes	Yes	Yes
AM	Yes	Yes	Yes	No	No	No
MAM	Yes	Yes	Yes	Controversial	No	No
E_{Deng}	Yes	No	No	No	No	No
E_{JS}	Yes	No	No	No	Yes	Yes
E_{PQ}	Yes	No	No	No	Yes	Yes
E_{inter}	Yes	No	No	No	No	Not proved

Table 3.1 summarizes which of the essential mathematical properties for uncertainty measures in ET, described in Section 3.3.1, are verified by each one of the uncertainty measure proposed so far in ET. A summary of the behavioral requirements satisfied by the uncertainty measures proposed so far in ET can be seen in Table 3.2. It can be deduced that, even though the maximum entropy involves a higher computational complexity than the other uncertainty measures, it is the only one that satisfies all essential mathematical properties, is coherently disaggregated into two measures that quantify conflict and non-specificity, and can be extended to more general theories than ET.

3.3.3 Uncertainty measures on belief intervals for singletons

Belief intervals for singletons have received considerable attention for quantifying the uncertainty-based information in ET in the last years. As pointed out in Section 2.2.4.4, the credal set associated with a BPA is always contained in the credal set compatible with the corresponding set of belief intervals for singletons. However, as shown in Example 2.2.1, there may be probability distributions consistent with the belief intervals for singletons but not with the BPA.

Nevertheless, belief intervals for singletons are more manageable than BPAs for representing uncertainty-based information. It is because, with the belief intervals for singletons, it is possible to know the uncertain area associated with each alternative, as shown in Figure 3.1. It does not happen directly

Table 3.2: Summary of the crucial behavioral requirements satisfied by the uncertainty measures proposed so far in ET. Complexity = computational complexity of the uncertainty measure; Disaggregation = whether the uncertainty measure is disaggregated into two measures that quantify conflict and non-specificity and the disggregation is coherent; Sensitivity = whether the uncertainty measure is sensitive to changes in the evidence, directly or via its parts of conflict and non-specificity; Extensible = whether there exists extension of the uncertainty measure to more general theories than ET.

UM	Complexity	Disaggregation	Sensitivity	Extensible
S^*	High	Coherent	Yes	Yes
AM	Low	No	Yes	No
MAM	Low	No	Yes	No
E_{Deng}	Low	Not very coherent	Yes	No
E_{JS}	Low	Not very coherent	Yes	No
E_{PQ}	Low	Not very coherent	Yes	No
E_{inter}	Low	Not very coherent	Yes	No

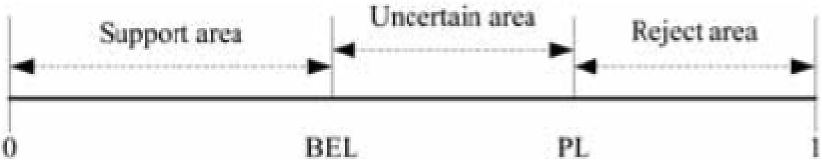

Figure 3.1: Uncertainty-based-information with belief intervals for singletons.

using the BPA. When employing the BPA, it is necessary to deal with the belief value of each subset, and we must remark that the number of subsets exponentially grows as the number of alternatives increases. Hence, several uncertainty measures on the belief intervals for singletons in ET have been proposed during the last years.

Let m be a BPA on X and Bel_m and Pl_m the belief and plausibility functions corresponding to m, respectively. Let $\mathcal{I}_m$ denote the set of belief intervals for singletons associated with m, determined via Equation (2.41). As demonstrated by Wang and Song [212], $\mathcal{I}_m$ is always reachable. Therefore, belief intervals for singletons are particular cases of reachable probability intervals.

We show below the total uncertainty measures proposed so far on $\mathcal{I}_m$.

- The total uncertainty measure defined by Yang and Han [223], $\mathrm{TUM}^{I}(\mathcal{I}_m)$, utilizes the following distance measure for intervals:

$$d^{I}\left([a_1, b_1], [a_2, b_2]\right) = \sqrt{\left[\frac{a_1 + b_1}{2} - \frac{a_2 + b_2}{2}\right]^2 + \frac{1}{3}\left[\frac{b_1 - a_1}{2} - \frac{b_2 - a_2}{2}\right]^2}. \tag{3.19}$$

TUM^{I} is defined as follows:

$$\mathrm{TUM}^{I}(\mathcal{I}_m) = 1 - \frac{\sqrt{3}}{t} \sum_{i=1}^{t} d^{I}\left([Bel_m\left(\{x_i\}\right), Pl_m\left(\{x_i\}\right)], [0, 1]\right). \tag{3.20}$$

In the previous expression, $\sqrt{3}$ is a normalization factor. The idea of this uncertainty measure is that a belief interval for a certain singleton is more uncertain as its distance to the interval $[0, 1]$, the one associated with total uncertainty, is lower.

Yang and Han [223] showed that the range of TUM^{I} is $[0, 1]$; it satisfies the monotonicity property; even though $\mathrm{TUM}^{I}(\mathcal{I}_m)$ is neither probabilistic nor set consistent, it has a rational behavior in many cases.

- In [220], a total uncertainty measure on belief intervals for singletons was developed to solve some drawbacks of the previous one. Such a measure employs the following distance function for intervals:

$$d^{I}_{E}\left([a_1, b_1], [a_2, b_2]\right) = \sqrt{(a_1 - a_2)^2 + (b_1 - b_2)^2}. \tag{3.21}$$

The uncertainty measure introduced in [220] is defined in the following way:

$$\mathrm{TUM}^{I}_{E}(\mathcal{I}_m) = \sum_{i=1}^{t} \left[1 - d^{I}_{E}\left([Bel_m\left(\{x_i\}\right), Pl_m\left(\{x_i\}\right)], [0, 1]\right)\right]. \tag{3.22}$$

In [220], it was shown via numerical examples that, when there are two identical BPAs on sets of alternatives with different cardinality, TUM^{I} may produce identical results, which implies a drawback. It does not occur with TUM$^{I}_{E}$.

- The total uncertainty measure proposed by Wang and Song [212] is defined as follows:

$$SU\left(\mathcal{I}_m\right) = \sum_{i=1}^{t}\left[-\frac{Bel_m\left(\{x_i\}\right)+Pl_m\left(\{x_i\}\right)}{2}\log_2\frac{Bel_m\left(\{x_i\}\right)+Pl_m\left(\{x_i\}\right)}{2}\right] + \sum_{i=1}^{t}\left[\frac{Pl_m\left(\{x_i\}\right)-Bel_m\left(\{x_i\}\right)}{2}\right]. \tag{3.23}$$

In Equation (3.23), the first term quantifies conflict, whereas the second term indicates non-specificity. In this way, the conflict value is quantified by means of the relation between the central values of the belief intervals for singletons, and the non-specificity value is determined by the span of such intervals [212].

As shown by Wang and Song [212], SU satisfies probabilistic consistency. When there exists $A \subseteq X$ such that $m(A) = 1$, then $SU\left(\mathcal{I}_m\right) = |A|$. Consequently, SU does not verify set consistency. However, the uncertainty of a classical set is determined by its cardinality. Therefore, SU satisfies a weak version of the set consistency property as, when $m(A) = 1$ for some $A \subseteq X$, it takes the form of an increasing function of $|A|$. The range of SU is equal to $[0, 1]$. Moreover, in [212], it was shown through numerical examples that SU might lead to more intuitive results than other uncertainty measures such as AM.

3.4 Uncertainty measures in more general imprecise probability theories

In addition to ET, the only imprecise probability theory in which uncertainty measures have been developed is credal sets. Hence, in this section, we focus on uncertainty measures on credal sets. The study of uncertainty measures in credal sets has its origin in the study of uncertainty measures in ET. Remark that, in most of the general imprecise probability theories, the uncertainty-based information can be represented by means of a credal set. The start point is that, in general credal sets, there also exists two types of uncertainty: conflict and non-specificity [13, 40].

Let $\mathcal{P}$ be a credal set on X and $\underline{P}$ the coherent lower probability function extracted from $\mathcal{P}$, computed via Equation (2.15).

The generalized Hartley measure GH, which is the well-established non-specificity measure in ET, can be extended to general credal sets. Such an extension is defined as follows [12]:

$$\mathrm{GH}(\mathcal{P}) = \sum_{A \subseteq X} m(A) \log_2(|A|), \tag{3.24}$$

where m is the Möbius inverse of $\underline{P}$.

Abellán and Moral [12] demonstrated that GH satisfies the following desirable properties for a non-specificity measure on credal sets:

1. It always holds that $\mathrm{GH}(\mathcal{P}) \geqslant 0$. Thus, GH is **well-defined** as a non-specificity measure.

2. If $\mathcal{P}$ contains a unique probability distribution, then $\mathrm{GH}(\mathcal{P}) = 0$.

3. **Monotonicity**: If $\mathcal{P}_1$ and $\mathcal{P}_2$ are two credal sets on X such that $\mathcal{P}_1 \subseteq \mathcal{P}_2$, then $\mathrm{GH}(\mathcal{P}_1) \leqslant \mathrm{GH}(\mathcal{P}_2)$.

4. The **range** of GH is equal to $[0, \log_2(t)]$. It reaches its maximum value, $\log_2(t)$, when $\mathcal{P}$ contains all probability distributions on X.

5. **Additivity**: Let X and X be two finite sets and $\mathcal{P}$ a credal set on the product space $X \times Y$. Let $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$ denote the marginal credal sets of $\mathcal{P}$ on X and Y, determined via Equations (2.6) and (2.7), respectively. Suppose that there is strong independence under $\mathcal{P}$ (See Definition 2.2.8). Then, the following equality is verified:

$$\mathrm{GH}(\mathcal{P}) = \mathrm{GH}(\mathcal{P}^{\downarrow X}) + \mathrm{GH}(\mathcal{P}^{\downarrow Y}).$$

The maximum entropy on $\mathcal{P}$, proposed by Abellán and Moral [13], is the well-established total uncertainty measure on a credal set. Such a measure is defined in the following way:

$$S^*(\mathcal{P}) = \max_{p \in \mathcal{P}} S(p), \tag{3.25}$$

$S(p)$ being the Shannon entropy of the probability distribution p, determined via Equation (3.2).

As pointed out in [40, 128], S^* is a well-established total uncertainty measure on credal sets since it satisfies a set of desirable properties. Specifically, Abellán and Moral [13] showed that S^* satisfies the following required properties for uncertainty measures on credal sets:

1. It is **well-defined** as it is always satisfied that $S^*(\mathcal{P}) \geqslant 0$.

2. The **range** of S^* is equal to $[0, \log_2(t)]$. $S^*(\mathcal{P})$ reaches the value 0 if, and only if, $\mathcal{P}$ only contains a probability distribution p such that $p(x_i) = 1$ for some $i \in \{1, 2, \dots, t\}$. When $\mathcal{P}$ contains the uniform probability distribution on X, the maximum value of S^*, $\log_2(t)$, is attained.

3. **Monotonicity**: Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be two credal sets on X satisfying $\mathcal{P}_1 \subseteq \mathcal{P}_2$. Then, it always holds that $S^*(\mathcal{P}_1) \leqslant S^*(\mathcal{P}_2)$.

4. **Subadditivity**: Let X and Y be two finite sets and $\mathcal{P}$ a credal set on the product space $X \times Y$. Let $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$ denote, respectively, the marginal credal sets of $\mathcal{P}$ on X and Y. It is always satisfied that:
$$S^*\left(\mathcal{P}\right) \leqslant S^*\left(\mathcal{P}^{\downarrow X}\right) + S^*\left(\mathcal{P}^{\downarrow Y}\right).$$

5. **Additivity**: If $\mathcal{P}$ is a credal set defined on a product space $X \times Y$ such that its marginal credal sets on X and Y, denoted respectively by $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$, are strongly independent under $\mathcal{P}$, then the following equality holds:
$$S^*\left(\mathcal{P}\right) = S^*\left(\mathcal{P}^{\downarrow X}\right) + S^*\left(\mathcal{P}^{\downarrow Y}\right).$$

Since it is supposed that in credal sets there are also two types of uncertainty, conflict and non-specificity, it must be possible to decompose S^* into two measures that respectively quantify conflict and non-specificity.

Abellán and Moral [15] proposed the following disaggregation for S^*:

$$S^*\left(\mathcal{P}\right) = S_*\left(\mathcal{P}\right) + \left[S^*\left(\mathcal{P}\right) - S_*\left(\mathcal{P}\right)\right], \tag{3.26}$$

where $S_*\left(\mathcal{P}\right)$ is the minimum entropy on $\mathcal{P}$:

$$S_*\left(\mathcal{P}\right) = \min_{p \in \mathcal{P}} S(p). \tag{3.27}$$

In Equation (3.26), the first term indicates conflict while the second one captures non-specificity.

The conflict measure S_* satisfies the following properties [15]:

1. The range of S_* is equal to $[0, \log_2(|X|)]$. $S_*\left(\mathcal{P}\right)$ is equal to 0 if, and only if, $\exists p \in \mathcal{P}$ such that $p(x_i) = 1$ for some $i \in \{1, 2, \dots, t\}$. $S_*\left(\mathcal{P}\right)$ attains its maximum value, $\log_2(|X|)$, when $\mathcal{P}$ just contains the uniform probability distribution.

2. S_* is **monotonously decreasing**. Formally, if $\mathcal{P}_1$ and $\mathcal{P}_2$ are two credal sets on X such that $\mathcal{P}_1 \subseteq \mathcal{P}_2$, then $S_*\left(\mathcal{P}_2\right) \leqslant S_*\left(\mathcal{P}_1\right)$.

3. **Continuity**: Small changes in $\mathcal{P}$ produce small changes in $S_*(\mathcal{P})$.

4. **Additivity**: Let $\mathcal{P}$ be a credal set on a product space $X \times Y$, X and Y being two finite sets. Let $\mathcal{P}^{\downarrow X}$ and $\mathcal{P}^{\downarrow Y}$ denote the marginal credal sets of $\mathcal{P}$ on X and Y, respectively. Suppose that there is strong independence under $\mathcal{P}$. Then, the following equality is satisfied:

$$S_*(\mathcal{P}) = S_*\left(\mathcal{P}^{\downarrow X}\right) + S_*\left(\mathcal{P}^{\downarrow Y}\right).$$

Regarding the non-specificity measure $S^* - S_*$, it verifies the following crucial properties for non-specificity measures on credal sets [15]:

1. The **range** of $S^* - S_*$ is equal to $[0, \log_2(|X|)]$. Its minimum value is attained when $\mathcal{P}$ is composed of a single probability distribution. When all probability distributions on X belongs to $\mathcal{P}$, $S^* - S_*$ reaches its maximum value, $\log_2(|X|)$.

2. $S^* - S_*$ is a **continuous** function, i.e, small changes in $\mathcal{P}$ lead to small changes in $S^* - S_*$.

3. It is **additive** as both S^* and S_* are additive.

4. $S^* - S_*$ is **increasing monotonous**. This means that, if $\mathcal{P}_1$ and $\mathcal{P}_2$ are two credal sets on X such that $\mathcal{P}_1 \subseteq \mathcal{P}_2$, then $(S^* - S_*)(\mathcal{P}_1) \leqslant (S^* - S_*)(\mathcal{P}_2)$.

Indeed, it is also possible to decompose S^* as follows:

$$S^*(\mathcal{P}) = GH(\mathcal{P}) + [S^*(\mathcal{P}) - GH(\mathcal{P})]. \tag{3.28}$$

The first term of Equation (3.28) captures non-specificity while the second one quantifies conflict. As demonstrated in [130], it always holds that $S^*(\mathcal{P}) - GH(\mathcal{P}) \geqslant 0$ and, thus, $S^* - GH$ makes sense as a conflict measure.

Abellán and Moral [15] argued that $S^* - S_*$ and GH have a different behavior as non-specificity measures: GH just measures absolute imprecision. In contrast, $S^* - S_*$ quantifies the imprecision by also taking the extreme probabilities into account. In this sense, $S^* - S_*$ has a more intuitive behavior than GH since the same absolute difference in probability values might be more important for probability distributions close to a degenerate one than for probability distributions close to the uniform distribution [15].

3.4.0.1 *Computation of the uncertainty measures in imprecise probabilities*

We have argued that the maximum entropy is a well-established total uncertainty measure on credal sets as it satisfies a set of required properties for this kind of measure. Moreover, this measure can be decomposed into two measures that coherently quantify conflict and non-specificity. Nonetheless, the calculation of the maximum entropy on a credal set might be very complex. Indeed, there is no an algorithm to compute the maximum entropy on an arbitrary credal set so far.

Some procedures have been proposed so far for computing the maximum entropy on special types of credal sets. For instance, several algorithms were developed to compute the maximum entropy on the credal set compatible with a BPA in ET [109, 118, 145, 154]. Abellán and Moral [16] proposed a procedure to compute the maximum entropy for Choquet capacities of order 2. Also, algorithms for obtaining the probability distribution of maximum entropy on reachable probability intervals and in some particular imprecise probability models have been proposed in the literature, which we describe below.

Regarding the non-specificity measures, the computation of GH tends to be pretty simple. In contrast, the computation of the minimum entropy is not trivial at all and, so far, there is no an algorithm for obtaining the minimum entropy on a general credal set. An algorithm to compute the minimum entropy on Choquet capacities of order 2 was proposed by Abellán and Moral [15]. We must remark that the computational complexity of such an algorithm is pretty high. Abellán [2] demonstrated a result that allows quickly obtaining the probability distribution that reaches the minimum entropy on an IDM credal set, which we detail below.

Maximum entropy on a reachable set of probability intervals Abellán and Moral [13] presented an algorithm for obtaining the probability distribution that reaches the maximum entropy on a reachable set of probability intervals. Such an algorithm have been widely used in practical applications where the uncertainty-based information is represented by means of probability intervals.

The mentioned algorithm aims to find a probability distribution consistent with the given set of intervals as close as possible to the uniform probability distribution. For this purpose, it starts by assigning, to each value of X, the lowest probability according to the given intervals. Then, a iterative procedure is carried out where, in each iteration, the probability of the values with the lowest probability is uniformly incremented considering the constraints imposed by the intervals and the second lowest probability value. The procedure finishes when the sum of all probability values is equal to 1 (a probability distribution is obtained).

Algorithm 3 exhaustively describes the procedure to obtain the probability distribution that attains the maximum entropy on a reachable set of probability intervals, where sec_min indicates the second minimum value. If such a second minimum value does not exist, then $sec\_min = -1$.

Algorithm 3: Procedure to compute the probability distribution of maximum entropy on a reachable set of probability intervals.

```
Procedure Determine probability distribution of maximum entropy
on a reachable set of probability intervals(Reachable set of probability
intervals on X $\{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$)
for $i = 1$ to $t$ do
    $\hat{p}(x_i) \leftarrow l_i$
$sum \leftarrow \sum_{i=1}^{t} \hat{p}(x_i)$
while $sum < 1$ do
    $min\_prob \leftarrow \min_{i \in \{1,2,\ldots,t\} | \hat{p}(x_i) < u_i} \hat{p}(x_i)$
    $index\_min\_prob \leftarrow \{i \in \{1, 2, \ldots, t\} \mid \hat{p}(x_i) = min\_prob\}$
    $num\_min \leftarrow |index\_min\_prob|$
    $sec\_min\_prob \leftarrow sec\_min_{i \in \{1,2,\ldots,t\} | \hat{p}(x_i) < u_i} \hat{p}(x_i)$
    for $i \in index\_min\_prob$ do
        if $sec\_min\_prob = -1$ then
            $\hat{p}(x_i) \leftarrow \hat{p}(x_i) + \min\left(u_i - \hat{p}(x_i), \frac{1-sum}{num\_min}, 1\right)$
        else
            $\hat{p}(x_i) \leftarrow \hat{p}(x_i) +$
              $\min\left(u_i - \hat{p}(x_i), sec\_min\_prob - min\_prob, \frac{1-sum}{num\_min}\right)$
    $sum \leftarrow \sum_{i=1}^{t} \hat{p}(x_i)$
return $\hat{p}$
```

Uncertainty measures on the Imprecise Dirichlet Model As pointed out in Section 2.3.1, a set of IDM probability intervals is always reachable. Hence, the maximum entropy on a set of IDM probability intervals can be computed by using Algorithm 3. Also, as said previously, Walley [209] suggests two values for the s parameter: $s = 1$ and $s = 2$. For $s \in [1, 2]$, Abellán [2] proposed a quick way of obtaining the maximum entropy on a set of IDM probability intervals.

Suppose that we have a sample of N independent and identically distributed outcomes of X. Let $n(x_i)$ denote the number of observations of x_i in the sample, $\forall i = 1, 2, \ldots, t$. Let us consider the set of values that have the minimum observed frequency:

$$\text{min\_observed\_IDM} = \left\{x_i \mid n(x_i) = \min_{j=1,2,\ldots,t} n(x_j)\right\}. \tag{3.29}$$

Let num_min_IDM denote the number of elements of X that have the minimum observed frequency, i.e, $\text{num\_min\_IDM} = |\text{min\_observed\_IDM}|$. In order to obtain the probability distribution that attains the maximum entropy with the IDM, $\hat{p}^{IDM}$, for $s \in [1, 2]$, two cases are distinguished [2]:

- **Case 1:** $\text{num\_min\_IDM} > 1$ **or** $s = 1$**:**

  In this case, for each $i = 1, 2, \ldots, t$, the value of the probability distribution of maximum entropy with the IDM is determined by:

$$\hat{p}^{IDM}(x_i) = \begin{cases} \frac{n(x_i) + \frac{s}{\text{num\_min\_IDM}}}{N+s} & \text{if} \quad x_i \in \text{min\_observed\_IDM} \\ \frac{n(x_i)}{N+s} & \text{if} \quad x_i \notin \text{min\_observed\_IDM} \end{cases}$$

- **Case 2:** $\text{num\_min\_IDM} = 1$ **and** $s > 1$**:**

  It implies that $\text{min\_observed\_IDM} = \{x_i\}$, for some $i \in \{1, 2, \ldots, t\}$. In this situation, we assign $n(x_i) \leftarrow n(x_i) + 1, \quad s \leftarrow s - 1$, recalculate the subset min_observed_IDM, and obtain $\hat{p}^{IDM}$ similarly to Case 1.

For obtaining the minimum entropy, since IDM probability intervals are reachable probability intervals and, therefore, Choquet capacities of order 2, the algorithm proposed by Abellán and Moral [15] to compute the minimum entropy in Choquet capacities of order 2 could be employed. However, Abellán [2] showed that, due to the special structure of IDM probability intervals, the minimum entropy with the IDM can be obtained in a straight way through the following result:

Theorem 3.4.1 *Let $(n(x_1), n(x_2), \ldots, n(x_t))$ be the array of observed frequencies in the sample. Let $(n_1^*, n_2^*, \ldots, n_t^*)$ denote the array of observed frequencies decreasingly ordered. Let $\underline{p} = \left(\underline{p}(x_1), \underline{p}(x_2), \ldots, \underline{p}(x_t)\right)$ be the probability distribution of minimum entropy with the IDM and $\underline{p}^* = \left(\underline{p}_1^*, \underline{p}_2^*, \ldots, \underline{p}_t^*\right)$ the same array decreasingly ordered. Such a probability distribution is determined as follows:*

$$\underline{p}^* = \left(\frac{n_1^* + s}{N+s}, \frac{n_2^*}{N+s}, \ldots, \frac{n_t^*}{N+s}\right).$$

The computation of the Hartley measure with the IDM is immediate, as the following result shows [2]:

Theorem 3.4.2 *Let $\mathcal{P}(\mathcal{I}_{IDM})$ be the IDM credal set associated with the sample, determined by means of Equation (2.63). The Hartley measure on such a credal set is obtained in the following way:*

$$GH\left(\mathcal{P}\left(\mathcal{I}_{IDM}\right)\right) = \frac{s}{N+s}\log_2(t).$$

Maximum entropy on the NPI-M As said in Section 2.3.2, the set of probability distributions compatible with the NPI-M is not convex. Thereby, the NPI-M is not representable via a credal set. In order to compute the maximum entropy with the NPI-M, it is necessary to manage with difficult constraints imposed by the probability wheel representation of the data employed in the NPI-M. The procedure to obtain the maximum entropy with the NPI-M was presented in [5]. As the mentioned procedure is notably complex, we do not detail it here.

In contrast, the approximate model A-NPI-M is representable by a reachable set of probability intervals. In consequence, for obtaining the maximum entropy with the A-NPI-M, Algorithm 3 can be utilized. Based on that algorithm, a more efficient procedure to compute the probability distribution of maximum entropy with the A-NPI-M was proposed in [5].

Suppose that there is a sample of N independent and identically distributed outcomes of X. Let $n(x_i)$ denote the number of observations in the sample for which $X = x_i, \quad \forall i = 1, 2, \dots, t$. Let t_{unobs} be number of unobserved values of X in the sample. Let t_1 denote the number of values of X observed once and t' the number of values observed at least twice. We shall denote T(i) the number of values of X observed i times:

$$T(i) = \left|\left\{x_j \mid n(x_j) = i, \quad 1 \leqslant j \leqslant t\right\}\right|. \tag{3.30}$$

Algorithm 4 [5] shows the procedure to obtain the probability distribution that attains the maximum entropy with the A-NPI-M.

Algorithm 4: Procedure to compute the probability distribution of maximum entropy with the A-NPI-M.

```
Procedure Determine probability distribution of maximum entropy with the A-NPI-M(Observed frequencies in the sample $(n(x_1), n(x_2), \ldots, n(x_t))$)
if $t' < t_{unobs}$ then
    for $j = 1$ to $t$ do
        if $n(x_j) \leqslant 1$ then
            $\hat{p}^{A-NPI-M}(x_j) \leftarrow \frac{t'+t_1}{N(t_{unobs}+t_1)}$
        else
            $\hat{p}^{A-NPI-M}(x_j) \leftarrow \frac{n(x_j)-1}{N}$
else
    for $j = 1$ to $t$ do
        if $n(x_j) \leqslant 1$ then
            $\hat{p}^{A-NPI-M}(x_j) \leftarrow \frac{1}{N}$
        else
            $\hat{p}^{A-NPI-M}(x_j) \leftarrow \frac{n(x_j)-1}{N}$
    $i \leftarrow 1$
    $mass \leftarrow t' - t_{unobs}$
    while $mass > 0$ do
        if $T(i) + T(i+1) < mass$ then
            for $j = 1$ to $t$ do
                $\hat{p}^{A-NPI-M}(x_j) \leftarrow \hat{p}^{A-NPI-M}(x_j) + \frac{1}{N}$
                $mass \leftarrow mass - 1$
        else
            for $j = 1$ to $t$ do
                $\hat{p}^{A-NPI-M}(x_j) \leftarrow \hat{p}^{A-NPI-M}(x_j) + \frac{mass}{N(T(i)+T(i+1))}$
            $mass \leftarrow 0$
        $i \leftarrow i + 1$
return $\hat{p}^{A-NPI-M}$
```

4 TRADITIONAL CLASSIFICATION

4.1 Introduction

Nowadays, in many domains, it is needed to deal with large amounts of data to make decisions. For this reason, *Data Science* [44] is essential. It is an interdisciplinary field that involves methods, processes, and systems from extracting knowledge for data and understanding better them.

Within Data Science, *classification* [107] is considered as an essential area. It consists of learning a model that , for a given instance described via a set of attributes or predictive features, predicts the value of a variable under study, also called the *class variable*. Classification has been widely employed in many domains. For example, in *medicine*, it is commonly used to predict whether a patient has a disease from a set of attributes of that patient; in *credit scoring*, it makes much sense to predict whether a loan should be given to a client from a set of features of that client; in *marketing*, given a set of attributes of a customer, classification is usually employed for predicting a preference of such a customer.

Many approaches to classification have been developed so far. Among such approaches, we can mention *Decision Trees* [177], *Nearest Neighbors* [232], *Bayesian Networks* [172], or *Artificial Neural Networks* (ANN) [229]. One of the most simple classification methods is the *Naïve Bayes algorithm* [85], which assumes that all predictive attributes are independent given the class variable. Despite this unrealistic assumption, Naïve Bayes has achieved good results in practice, often comparable with more sophisticated classification algorithms, especially when the predictive attributes are not highly correlated [82, 93, 133]. Also, Decision Trees are known to be very simple, efficient, transparent, and interpretable models. Moreover, *ensembles of classifiers*, which consider many individual classifiers and combine their predictions to give a final one, often perform better than single classifiers even though the computational complexity of ensemble methods is much higher. In this thesis work, we consider Nearest Neighbors, Naïve Bayes, Decision Trees, and ensemble algorithms.

The mentioned classification algorithms often use a mathematical model to represent the uncertainty-based information about the class variable involved in a classification dataset. Most of the classification algorithms proposed so far employ classical probability theory for this purpose. Decision Trees based on imprecise probabilities were proposed few years ago, which are known as *Credal Decision Trees* (CDTs) [1]. CDTs have obtained better results than Decision Trees that use precise probabilities, the improvement being more notable as there is more class noise[1] in the data [148–150].

Classifiers usually aim to minimize the number of instances incorrectly classified. This issue would be optimal if all classification errors had the same importance. Nevertheless, in practical applications, classification errors often have different costs. For example, in *medical diagnosis*, the cost of incorrectly predicting that a patient does not have a serious disease may be much higher than the cost of erroneously predicting that the patient is ill [140, 171, 183]; in *credit fraud detection*, predicting that a credit card is legal when it is fraudulent is likely to cause far higher economical losses for banks and financial institutions than predicting a normal credit card as fraudulent [24, 166, 182]; in *software defect prediction*, the cost of non-defective modules predicted as defective is probably far lower than the cost of defective modules predicted as non-defective [26, 142, 191]. For this reason, classifiers that take the costs of errors into account, called *cost-sensitive classifiers*, have been developed [89].

This chapter is organised as follows: Section 4.2 describes the classification problem in detail. In Section 4.3, the Nearest-Neighbors algorithm is detailed. The Naive Bayes algorithm is explained in Section 4.4. Classical Decision Trees and Decision Trees based on imprecise probabilities are described in Sections 4.5 and 4.6, respectively. In Section 4.7, ensembles of classifiers are introduced. The cost-sensitive classification problem is summarized in Section 4.8.

4.2 Classification paradigm

The classification task consists of learning a model which, for a given instance described by means of a set of predictive attributes or features, predicts the value of the class variable corresponding to such an instance.

Formally, the classification problem starts from the following issues:

- A set of d predictive attributes $\{X^1, X^2, \dots, X^d\}$. Let $\mathrm{Dom}(X^i)$ denote the domain of the X^i attribute, $\forall i = 1, 2, \dots, d$.
- A class variable C whose set of possible values is $\Omega_C = \{c_1, c_2, \dots, c_K\}$, with $K \geqslant 2$.

[1] The term ‘noise’ is used to indicate that there are errors in the data. In particular, ‘class noise’ corresponds to errors in the class value for some instances.

The goal of classification is to learn a function $h : \left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right) \rightarrow \Omega_C$ that, for a given instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in Dom(X^i) \quad \forall i = 1, 2, \ldots, d$, returns the predicted class value for that instance, namely $h\left(\mathbf{x}\right)$. The learned model can also be determined by a real-valued function $f : \left(Dom(X^1), Dom(X^2), \ldots\right.$ $\Omega_C \rightarrow \mathbb{R}$ which, for a given instance an a class value $c_j \in \Omega_C$, returns the predicted posterior probability that the class value of the instance is c_j.

For learning the classification model, a training set $\mathcal{D}_{tr}$ is usually employed, where each instance in $\mathcal{D}_{tr}$ is described by a set of attribute values and has a unique value of the class variable.

4.2.1 Evaluation of a classifier

To evaluate the performance of a classification algorithm described by means of a model $h : \left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right) \rightarrow C$, a test set different from the set utilized for training, $\mathcal{D}_{test}$, is commonly used.

The most common evaluation measure in Classification is *Accuracy*. It consists of the proportion of test instances correctly classified.

Formally, let $N_{test} = |\mathcal{D}_{test}|$ be the number of test instances. Let $x^i_{r_{ij}} \in Dom(X^i)$ denote the value of the ith attribute for the jth test instance, $\forall j = 1, 2, \ldots, N_{test}$, $i = 1, 2, \ldots, d$, $\mathbf{x_j} = \left(x^1_{r_{1j}}, x^2_{r_{2j}}, \ldots, x^d_{r_{dj}}\right)$, and c^j the class value of the jth test instance, with $c^j \in \{c_1, c_2, \ldots, c_K\}, \quad \forall j = 1, 2, \ldots, N_{test}$. Accuracy is determined as follows:

$$Accuracy(h) = \frac{1}{N_{test}} \sum_{j=1}^{N_{test}} \left[\left[h(\mathbf{x_j}) = c^j\right]\right], \tag{4.1}$$

$[[\cdot]]$ being an indicator function, which takes the value 1 if the condition is satisfied and 0 otherwise.

We may note that this metric assumes that all classification errors have the same importance. However, when the class variable is binary, i.e, it takes values in $\{0, 1\}$, there are usually much more instances satisfying $C = 0$ than instances for which $C = 1$ but the cost of incorrectly classifying instances belonging to the latter group tends to be higher than the cost of erroneously classifying instances that belong to the former group. In these cases, the Accuracy measure is not the best one and, instead, other evaluation metrics are often utilized [95].

Assuming that C is binary, let $TP(h)$ and $TN(h)$ denote the number of test instances correctly classified via h that satisfy $C = 1$ and $C = 0$, respectively.

Likewise, let FP and FN be the number of misclassified instances through h for which $C = 1$ and $C = 0$, respectively. The following evaluation measures are commonly employed in binary classification:

- **Precision**: It indicates, between the test instances for which $C = 1$ is predicted, the proportion of them correctly classified:

$$\mathrm{Precision}(h) = \frac{TP(h)}{TP(h) + FP(h)}. \tag{4.2}$$

- **Recall**: It is the proportion of positive test instances correctly classified:

$$\mathrm{Recall}(h) = \frac{TP(h)}{TP(h) + FN(h)}. \tag{4.3}$$

- **F1**: It is the harmonic mean between Precision and Recall:

$$F1(h) = \frac{2 \cdot \mathrm{Precision}(h) \cdot \mathrm{Recall}(h)}{\mathrm{Precision}(h) + \mathrm{Recall}(h)}. \tag{4.4}$$

For binary classification problems where the number of positive instances is much higher than the number of negative instances but the cost of missclassifying positive instances is considerably higher than the cost of incorrectly classifying negative instances, F1 is an appropriate evaluation metric as it measures a trade-off between recognizing positive instances and not predicting too many instances as positive.

In the mentioned binary classification problems, the **Receiver Operator Characteristic (ROC) curve** is also frequently employed to measure the performance of a binary classifier. The ROC curve represents the TP rate against the FP rate under different threshold values used in the real-valued function f to separate instances classified as positive and negative. The **Area Under ROC curve (AUC)** summarizes a ROC curve and is a useful evaluation metric for unbalanced binary classification problems where false negatives have worse consequences than false positives or vice-versa. Such an evaluation metric measures the ability of a classifier to distinguish between positive from negative instances.

4.2.1.1 *Cross-validation*

When it is wanted to validate the performance of a classifier in a dataset through evaluation metrics, a cross-validation procedure is commonly used in the literature for the results do not depend on the data utilized for training and testing.

A cross-validation procedure divides the given dataset into a fixed number of partitions, usually called the number of *folds*. For each partition, it does an iteration in which the corresponding partition is used as the test set and the rest of the data as the training set. In each iteration, the evaluation metrics are extracted with the test set. Finally, for each metric, the average value across all iterations is computed.

4.2.1.2 *Sensitivity to noise*

In order to test the sensitivity to noise[2] of a classifier, the *Equalized Loss Accuracy* metric (ELA) [181] is suitable to be employed. Such a metric, for given noise level in the data, measures how differ the accuracy a classifier with that noise level from the total accuracy, normalizing by the accuracy of the classifier without noise in the data.

Formally, let $nois\_lev$ be the level of noise in the data. Let h denote the learned classification model and $Acc_0(h)$ and Acc_{nois_lev} the accuracy of the classifier without noise and with $nois\_lev$% of noise, respectively. For the level noise $nois\_lev$, ELA is defined as follows:

$$\mathrm{ELA}_{nois\_lev}(h) = \frac{1 - Acc_{nois\_lev}}{Acc_0(h)}. \tag{4.5}$$

This metric presents some advantages over other measures of sensitivity of classifiers to noise. For more details, see [181].

4.2.1.3 *Statistical comparisons between classification methods*

To compare the performance of multiple classification methods on many datasets according to any of the metrics described above, the recommendations given by Demšar [75] tend to be used. Such recommendations can be summarized in the following way:

- When there are only two algorithms to compare, the **Wilcoxon test** [216] should be used. It is a non-parametric test that ranks the absolute values of the differences in the performance of the algorithms across all datasets. Then, it computes the sum of the ranks of the positive and negative differences. The null hypothesis of this test is that both algorithms perform equivalently.

  Formally, let val_i^1 and val_i^2 be, respectively the values obtained by the first and the second classifier in the ith dataset, $\forall i = 1, 2, \ldots, n\_dat$,

[2] Here, we only consider class noise.

n_dat being the number of datasets used in the statistical comparison. Let d_i be the normalized difference between the values obtained by both algorithms in the ith dataset:

$$d_i = \frac{val_i^1 - val_i^2}{val_i^1}, \quad \forall i = 1, 2, \ldots, n\_dat.$$

Let $rank(d_i)$ denote the rank of the absolute value of d_i, $\forall i = 1, 2, \ldots, n\_dat$. The Wilcoxon test is based on the following statistic:

$$W = \sum_{i=1}^{n\_dat} sign(val_i^1 - val_i^2) rank(d_i),$$

where, $\forall i = 1, 2, \ldots, n\_dat$:

$$sign(val_i^1 - val_i^2) = \begin{cases} 1 & \text{if} \quad val_i^1 > val_i^2 \\ -1 & \text{if} \quad val_i^1 < val_i^2 \\ 0 & \quad otherwise \end{cases}$$

According to the Wilcoxon test, W has a distribution of mean 0 and variance $\frac{n\_dat(n\_dat+1)(2n\_dat+1)}{6}$.

- For comparing the results obtained by three or more algorithms, it is recommended to employ the **Friedman test** [92]. Such a test is non-parametric and separately ranks the algorithms for each dataset. Let $rank_i^j$ be the rank obtained by the jth algorithm for the ith dataset, $\forall i = 1, 2, \ldots, n\_dat, \quad j = 1, 2, \ldots, n\_alg$, n_alg being the number of algorithms to compare. Let $Rank(j)$ denote the sum of the ranks obtained by the jth algorithm across all datasets:

$$Rank(j) = \sum_{i=1}^{n\_dat} rank_i^j, \quad \forall j = 1, 2, \ldots, n\_alg.$$

The Friedman test is based on the following statistic:

$$\chi_F^2 = \frac{12n\_dat}{n\_alg\,(n\_alg+1)} \left[\sum_{j=1}^{n\_alg} Rank(j)^2 - \frac{n\_alg\,(n\_alg+1)^2}{4} \right].$$

χ_F^2 has a chi square distribution with n_alg - 1 degrees of freedom. The null hypothesis of the Friedman test is that all algorithms perform equivalently.

When the null hypothesis of the Friedman test is rejected, the following tests are commonly used for comparing the algorithms pairwise.

- **Nemenyi test** [167]: According to this test, there are statistically significant differences between two algorithms if, and only if, the difference between their average Friedman ranks is lower than the following critical distance:

$$CD = q_{\alpha}\sqrt{\frac{n\_alg\,(n\_alg+1)}{6n\_dat}}, \tag{4.6}$$

  where q_{α} is a critical value based on the Studentized ranged divided by $\sqrt{2}$. The level of significance considered for the critical distance is equal to the original level of significance, α, divided by the number of pairwise comparisons, $n\_comp = \frac{n\_alg(n\_alg-1)}{2}$, that is, $\frac{\alpha}{n\_comp}$.
- **Holm test** [115]: This test decreasingly orders the differences between the Friedman ranks between pairs of algorithms. As the Nemenyi test, the Holm test is also based on critical distances between pair of algorithms, computed via Equation (4.6). However, while the Nemenyi test uses the same level of significance for all pairs of algorithms, the Holm test carries out the following iterative procedure: The level of significance considered for the critical distance between the algorithms that have the highest difference of Friedman ranks is equal to $\frac{\alpha}{n\_comp}$. If the difference of the Friedman ranks is lower than such a critical distance, then all pairs of algorithms perform equivalently. Otherwise, there are statistically significant differences between that pair of algorithms, and the critical distance for the algorithms with the second-highest difference of Friedman ranks is computed with a level of significance of $\frac{\alpha}{n\_comp-1}$. If the difference is lower than the critical distance, then there are no statistically significant differences between these two algorithms. Otherwise, the algorithm with the lower Friedman rank significantly outperforms the other one, and the process is iteratively repeated until a pair of algorithms that perform equivalently is found, or all pairs of algorithms are checked.

In order to represent the results of the Friedman and Nemenyi (Holm) tests, critical diagrams [75] tend to be used. A critical diagram utilizes an enumerated axis for drawing the average Friedman ranks of the classifiers. The algorithms are arranged so that the ones with the highest rank are placed at the right-most side. Segments are used to connect the algorithms for which there are no statistically significant differences according to the Nemenyi (Holm) test.

Moreover, the **Paired t-test** is sometimes used for comparing the performance of two algorithms over a cross-validation procedure repeated several times over a single dataset. The null hypothesis of this test is that the difference between the means values obtained by both algorithms is equal to 0.

Formally, let n_tests be the number of test sets obtained in a cross-validation procedure repeated several times on a dataset. Let val_i^1 and val_i^2 denote, respectively, the results obtained by the first and second algorithm in the ith test set. Let $diff_i$ be the difference between the results obtained by the first and second algorithm in the ith dataset, that is, $diff_i = val_i^1 - val_i^2, \quad \forall i = 1, 2, \ldots, n\_tests$. Let $avg(diff)$ and $sigma(diff)$ be, respectively, the mean and standard deviation of such differences. The following t-statistic is considered in the Paired t-test:

$$t\_statistic = \frac{avg(diff)}{sigma(diff)}.$$

According to the Paired t-test, $t\_statistic$ is distributed with a Student distribution with $n\_tests - 1$ degrees of freedom.

4.3 Nearest-Neighbors algorithm

The Nearest Nearest Neighbors algorithm (NN) [64] is a lazy approach to classification in the sense that it does not carry out a training phase.

Suppose that it is required to classify an instance whose attribute vector is $\mathbf{x} = \left(x_{r_1}^1, x_{r_2}^2, \ldots, x_{r_d}^d\right)$, where $x_{r_i}^i \in Dom(X^i) \quad \forall i = 1, 2, \ldots, d$. NN computes the $nun\_neighbors$-nearest neighbors of the instance by using a distance function on the attribute space. For each class value $c_j \in \Omega_C$, let $\mathcal{K}_j(\mathbf{x})$ denote the number of neighbors of $\mathbf{x}$ (among the $num\_neighbors$-nearest ones) whose class value is c_j. NN predicts for $\mathbf{x}$ the class value with the highest number of neighboring instances associated with it, that is,

$$h^{NN}(\mathbf{x}) = \text{arg}\_max_{j \in \{1,2,\ldots,K\}} \mathcal{K}_j(\mathbf{x}). \tag{4.7}$$

4.4 Naive Bayes

Hereon, we suppose that the domain of each attribute X^i attribute is a finite set, namely $Dom(X^i) = \left\{x_1^i, x_2^i, \ldots, x_{t_i}^i\right\}, \quad \forall i = 1, 2, \ldots, d$.

The Naive Bayes algorithm (NB) [85] is based on the naive assumption [85], which states that all attributes are independent given the class variable. This means that, $\forall j = 1,2,\dots,K, \quad r_i = 1,2,\dots,t_i, \quad i = 1,2,\dots,d$:

$$P(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d} \mid C = c_j) = \prod_{i=1}^{d} P(X^i = x^i_{r_i} \mid C = c_j). \quad (4.8)$$

Suppose that it is required to classify an instance for which $X^i = x^i_{r_i}$, with $r_i \in \{1,2,\dots,t_i\} \quad \forall i = 1,2,\dots,d$. The NB model predicts the class value $c_k \in \Omega_C$ that verifies:

$$c_k = \arg \max_{j=1,2,\dots,K} P(C = c_j \mid X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}). \quad (4.9)$$

Bayes theorem leads to:

$$\begin{aligned} &P(C = c_j \mid X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}) = \\ &\frac{P(C = c_j, X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d})}{P(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X_d = x^d_{r_d})} = \\ &\frac{P(C = c_j) P(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d} \mid C = c_j)}{P(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X_d = x^d_{r_d})}, \quad \forall j = 1,2,\dots,K. \end{aligned}$$

In this way, we may deduce that

$$\begin{aligned} &\arg \max_{j=1,2,\dots,K} P(C = c_j \mid X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}) = \\ &\arg \max_{j=1,2,\dots,K} P(C = c_j) P(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d} \mid C = c_j) = \\ &\arg \max_{j=1,2,\dots,K} P(C = c_j) \prod_{i=1}^{d} P(X^i = x^i_{r_i} \mid C = c_j), \end{aligned}$$

where the last equality is a consequence of the naive assumption (Equation (4.8)).

In addition, due to Bayes theorem, it holds that:

$$P(X^i = x^i_{r_i} \mid C = c_j) = \frac{P(X^i = x^i_{r_i}, C = c_j)}{P(C = c_j)} = \frac{P(C = c_j \mid X^i = x^i_{r_i}) P(X^i = x^i_{r_i})}{P(C = c_j)},$$

$\forall i = 1,2,\dots,d, \quad j = 1,2,\dots,K$.

Consequently, to classify an instance whose attribute vector is $\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d})$, with $r_i \in \{1,2,\dots,t_i\}, \quad \forall i = 1,2,\dots,d$, the NB algorithm predicts the following class value:

$$h^{NB}(\mathbf{x}) = \arg \max_{c_j \in \{c_1, c_2, \dots, c_K\}} P(C = c_j) \prod_{i=1}^{d} \frac{P(C = c_j \mid X^i = x^i_{r_i})}{P(C = c_j)}. \quad (4.10)$$

4.4.1 Estimation of the probabilities in Naive Bayes

The key issue of the NB algorithm is the estimation of the probabilities $\hat{P}(C = c_j)$ and the conditional probabilities $\hat{P}(C = c_j \mid X^i = x^i_{r_i}), \ \forall j = 1, 2, \ldots, K, \ \ r_i = 1, 2, \ldots, t_i, \ \ i = 1, 2, \ldots, d$. Let N_{tr} be the number of instances in the training set. Let $n_{tr}(c_j)$ denote the number of training instances that satisfy $C = c_j$, $n_{tr}(x^i_{r_i})$ the number of training instances for which $X^i = x^i_{r_i}$, and $n_{tr}(x^i_{r_i,j})$ the number of training instances that verify $X^i = r_i \wedge C = c_j, \ \ \forall r_i = 1, 2, \ldots, t_i, \ \ i = 1, 2, \ldots, d, \ \ j = 1, 2, \ldots, K$.

The main approaches proposed so far to estimate the aforementioned probabilities in NB can be summarized as follows:

- **Classical estimations** are determined through relative frequencies in the training set:

  $$\hat{P}_{cla}(C = c_j) = \frac{n_{tr}(c_j)}{N_{tr}}, \quad \hat{P}_{cla}(C = c_j \mid X^i = x^i_{r_i}) = \frac{n_{tr}(x^i_{r_i,j})}{n_{tr}(x^i_{r_i})},$$
  $$\forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d, \quad j = 1, 2, \ldots, K.$$

  The main problem of classical estimators arises when $n_{tr}(x^i_{r_i,j}) = 0$ for some $i \in \{1, 2, \ldots, d\}$. In this case, $\hat{P}_{cla}(C = c_j \mid X^i = x^i_{r_i}) = 0$, which implies that $\hat{P}_{cla}(C = c_j) \prod_{i=1}^{d} \frac{\hat{P}_{cla}(C=c_j|X^i=x^i_{r_i})}{P_{cla}(C=c_j)} = 0$. Thus, in this situation, the value of $\hat{P}_{cla}(C = c_j \mid X^i = x^i_{r_i})$ decisively influences the probability value estimated for c_j since such a value can be equal to 0 even though $\hat{P}_{cla}(C = c_j$ and $\hat{P}_{cla}(C = c_j \mid X^{i'}_{r_{i'}})$ are pretty high $\forall i' \in \{1, 2, \ldots, d\} \setminus \{i\}$. Furthermore, when $n_{tr}(x^i_{r_i})$ is very small, the estimation of $P(C = c_j \mid X^i = x^i_{r_i})$ might be quite unstable, $\quad \forall i = 1, 2, \ldots, d$.

- **Laplace** law of succession [103] was introduced to solve the problem that appears in classical estimators when the frequency of an attribute value is equal to 0 or very small. It assumes an uniform prior distribution of all class values. Given the training set, the Laplace's estimations of the aforementioned probabilities are determined in the following way:

  $$\hat{P}_{Lap}(C = c_j) = \frac{n_{tr}(c_j) + 1}{N_{tr} + K}, \quad \hat{P}_{Lap}(C = c_j \mid X^i = x^i_{r_i}) = \frac{n_{tr}(x^i_{r_i,j}) + 1}{n_{tr}(x^i_{r_i}) + K},$$
  $$\forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d, \quad j = 1, 2, \ldots, K.$$

  In spite of alleviating the main drawback of classical estimators, Laplace's estimation also presents two shortcomings:

1. If $n_{tr}(x^i_{r_i}) = 0$ for some $i \in \{1, 2, \ldots, d\}$, then $\frac{\hat{P}_{Lap}(C=c_j|X^i=x^i_{r_i})}{\hat{P}_{Lap}(C=c_j)} = \frac{1}{K\hat{P}_{Lap}(C=C_j)}$. Thereby, the estimation of the conditional probability increases as $\hat{P}_{Lap}(C = c_j)$ decreases and vice-versa.
2. When $n_{tr}(x^i_{r_i,j}) = 0$ for some $i \in \{1, 2, \ldots, d\}$, $j \in \{1, 2, \ldots, K\}$, it holds that $\frac{\hat{P}_{Lap}(C=c_j|X^i=x^i_{r_i})}{\hat{P}_{Lap}(C=c_j)} = \frac{1}{\left(K+n_{tr}(x^i_{r_i})\right)\hat{P}_{Lap}(C=c_j)}$. Hence, in this situation, the estimation of the conditional probability is inversely proportional to $\hat{P}_{Lap}(C = C_j)$. This can be considered as a strange behavior due to the assumption of uniformity for the prior distribution of the class values.

- m-**probability estimation model**: In order to correct the questionable behavior of Laplace's estimation in the above-mentioned situations, a more appropriate and flexible set of prior probabilities was proposed in [102]. According to it, after $succ$ successes in N trials, the probability of getting a success in the next trial is equal to:

  $$\hat{P}(succ, N) = \frac{succ + a}{N + a + b}, \tag{4.11}$$

  where $a > 0$ and $b > 0$. We may note that Laplace's estimation is a particular case of Equation (4.11) where $a = 1$ and $b = K - 1$. Cestnik [47, 48] made the following choice of these parameters: $a = mP(C = c_j)$ and $b = m - a$, m being a parameter of the model. This estimation is called the m-*probability estimation*.

  In this way, Cestnik [47] proposed the following estimation for the conditional probabilities:

  $$\hat{P}_{Ces}(C = c_j \mid X^i = x^i_{r_i}) = \frac{n_{tr}(x^i_{r_i,j}) + m\hat{P}_{Lap}(C = c_j)}{n_{tr}(x^i_{r_i}) + m}, \tag{4.12}$$
  $$\forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d, \quad j = 1, 2, \ldots, K.$$

  It can be observed that, for the estimation of the probabilities of the class values, $P(C = c_j) \quad \forall j = 1, 2, \ldots, K$, the NB model developed by Cestnik [47] employs the Laplace's law of succession. We may note that, in such a model, the probability estimated for the class value without the attribute values influences the estimation of the conditional probabilities.

  As pointed out by Cestnik [47], the value of the m parameter should be higher as there is more noise in the data.

4.5 Classical Decision Trees

Decision Trees [176] are based on a recursive partition procedure which, at each level, splits the training data according to the possible values of an attribute. Such an attribute is selected via a criterion based on the uncertainty-based information about the class variable at that level. In this way, in a Decision Tree, each node corresponds to an attribute and has a branch for each possible value of that attribute. When selecting an attribute to split at a level does not provide more uncertainty-based information about the class variable, or there are no more attributes to insert according to an established criterion, a *leaf* or *terminal* node is obtained. A class value is assigned to such a terminal node. Algorithm 5 summarizes the procedure to build a generic Decision Tree. The representation of the data involved in a Decision Tree is quite simple, transparent, and interpretable.

Algorithm 5: Generic procedure to build a Decision Tree.

```
Procedure Build_DT(Node $\mathcal{N}$)
Let $\mathcal{D}$ be the dataset associated with $\mathcal{N}$
if There are more attributes to insert then
    Select $X^i$ the attribute that leads to the maximum gain of
      information about the class variable on $\mathcal{D}$ according to a criterion.
    for $x^i_{r_i}$ possible value of $X^i$ do
        Make a node $\mathcal{N}_{r_i}$ child of $\mathcal{N}$
        Build_DT($\mathcal{N}_{r_i}$)
else
    Make $\mathcal{N}$ a leaf node
    Assing a class value to $\mathcal{N}$
```

When a Decision Tree built from a training set is very large, it tends to overfit the training data. For this reason, a post-pruning process is often utilized to remove branches that do not contribute to the generalization accuracy [147]. Several experimental studies have highlighted that post-pruning methods improve the performance of a Decision Tree, especially when there is noise in the data. Examples can be found in [52, 159].

Hence, the building process of a Decision Tree is principally determined by the following issues:

1. The criterion employed for selecting the attribute to split in each node, also known as the *split criterion*.

2. The conditions under which it is stopped branching the tree.

3. The criterion used to assign a class value to each terminal node.

4. The post-pruning process of the tree.

Among the previous points, the most important might be the split criterion. Let $\mathcal{D}$ be the subset of the training set associated with a certain node. The split criteria employed in classical Decision Trees are based on one of the two following uncertainty measures of class variable on $\mathcal{D}$:

- **Gini index** [39]: It measures the diversity of the class variable in $\mathcal{D}$:

$$\text{Gini}^{\mathcal{D}}(C) = 1 - \sum_{j=1}^{K} P^{\mathcal{D}}(C = c_j)^2, \tag{4.13}$$

  $P^{\mathcal{D}}(C = c_j)$ being the probability that $C = c_j$ in $\mathcal{D}$, estimated via the proportion of instances in $\mathcal{D}$ for which $C = c_j, \quad \forall j = 1, 2, \ldots, K$.

- **Shannon entropy** [189]:

$$S^{\mathcal{D}}(C) = -\sum_{j=1}^{K} P^{\mathcal{D}}(C = c_j) \log_2 \left(P^{\mathcal{D}}(C = c_j) \right). \tag{4.14}$$

Among the split criteria utilized in classical Decision Trees, the following ones are remarkable:

- **Gain Gini Index**: It is the split criterion used in the CART algorithm proposed by Breiman [39]. For each attribute X^i, it is determined by[3]:

$$GGI^{\mathcal{D}}(C, X^i) = \text{Gini}^{\mathcal{D}}(C) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) \text{Gini}^{\mathcal{D}}(C \mid X^i = x^i_{r_i}), \tag{4.15}$$

  where $P^{\mathcal{D}}(X^i = x^i_{r_i})$ is the probability that $X^i = x^i_{r_i}$ in $\mathcal{D}$, estimated through relative frequencies, and $\text{Gini}^{\mathcal{D}}(C \mid X^i = x^i_{r_i})$ is the Gini index of C on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}, \quad \forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

[3] Within this section, for presenting the split criteria, as in the previous section, we assume that the domain of each attribute is finite, that is, $\text{Dom}(X^i) = \{x^i_1, x^i_2, \ldots, x^i_{t_i}\}, \quad \forall i = 1, 2, \ldots, d$. However, most Decision Trees can handle continuous attributes with particular methods.

- **Info-Gain** (IG): It was proposed by Quilan [176] for ID3, the first Decision Tree developed. For each attribute X^i, IG is defined as follows:

$$IG^{\mathcal{D}}(C, X^i) = S^{\mathcal{D}}(C) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) S^{\mathcal{D}}(C \mid X^i = x^i_{r_i}), \tag{4.16}$$

  $S^{\mathcal{D}}(C \mid X^i = x^i_{r_i})$ being the Shannon entropy of C on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}, \quad \forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

- **Info-Gain Ratio** (IGR): It was introduced by Quilan [177]. For each attribute X^i, IGR is defined in the following way:

$$IGR^{\mathcal{D}}(C, X^i) = \frac{IG^{\mathcal{D}}(C, X^i)}{S^{\mathcal{D}}(X^i)}, \tag{4.17}$$

  where $S^{\mathcal{D}}(X^i)$ is the Shannon entropy of X^i on $\mathcal{D}$:

$$S^{\mathcal{D}}(X^i) = -\sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) \log_2 \left(P^{\mathcal{D}}(X^i = x^i_{r_i}) \right). \tag{4.18}$$

For classifying an instance with a Decision Tree, a path from the root node to a leaf node is made by using the attribute values of that instance. Then, the class value predicted for that instance is the one assigned to that terminal node. The procedure to classify an instance with a Decision Tree is summarized in Algorithm 6.

Algorithm 6: Procedure to classify an instance with a generic Decision Tree.

```
Procedure Classify_DT(Tree $\mathcal{T}$, instance with attribute vector
$\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}))$
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the
   attribute values $x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}$.
2. Assign the predicted class value at $\mathcal{L}$ to $h^{DT}(\mathbf{x})$.
return $h^{DT}(\mathbf{x})$
```

4.5.1 C4.5 Decision Tree

The first Decision Tree algorithm was ID3, introduced by Quinlan [176]. ID3 uses IG as the split criterion. This split criterion facilitates the selection of attributes with many possible values. It must also be remarked that ID3 does not use any post-pruning process, which may lead to over-fitting. Moreover, ID3 works with neither continuous attributes nor missing values.

In order to solve these shortcomings, Quinlan [177] presented the C4.5 algorithm. This is probably the most-known classical Decision Tree. C4.5 is based on the following points:

- **Split criterion**: C4.5 uses IGR, unlike ID3, which employs IG. The latter split criterion boosts attributes with more possible values. However, IGR normalizes IG by the entropy of the attribute. Thus, IGR penalizes attributes with many possible values. For this reason, it can be stated that the split criterion of C4.5 is more suitable than the one of ID3. C4.5 selects the attribute with the highest IGR value whenever that value is higher than the average IGR value between the attributes that are numeric or whose number of possible values is lower than 0.3 times the number of instances in the corresponding branch.

- **Stop branching criterion**: C4.5 stops branching the tree when there is no attribute for which the IGR value is not equal to 0, or there are no attributes that are numeric or whose number of possible values is lower than 0.3 times the number of instances in the associated branch. Also, the stop branching criterion of C4.5 considers that there is a minimum number of instances per leaf, which is often set to 2. Consequently, C4.5 does not split the dataset via an attribute if there is a value such that the number of instances going down the corresponding branch it is lower than the established minimum.

- **Criterion for assigning class values to leaf nodes**: At each leaf node, C4.5 assigns the most frequent class value between the instances at that terminal node. Formally, at a leaf node $\mathcal{L}$, let $n^{\mathcal{L}}(c_j)$ denote the frequency of c_j at $\mathcal{L}$, $\quad \forall j = 1, 2, \ldots, K$. The class value assigned to that leaf node is determined by:

  $$c_k \mid k = \arg \max_{j=1,2,\ldots,K} n^{\mathcal{L}}(c_j). \tag{4.19}$$

  If there is a tie between two class values, namely c_i and c_j, then the first one of then is chosen, that is, c_i if $i < j$ and c_j if $j < i$. Ties between three or more class values are also broken by choosing the first one of them.

- **Numeric attributes**: For a numeric attribute, C4.5 just considers binary splits. It considers each split point, and chooses the one that gives rise to the maximum IGR value.

- **Missing values**: In order to deal with missing values, C4.5 considers instance weights. The initial weight of an instance is always equal to 1. When the value of the attribute corresponding to a node is missing for an instance, that instance goes down each branch with a weight equal to the proportion of instances at the branch. In these cases, it is necessary to adapt the IGR split criterion (Equation (4.17)) for working with proportions of weights rather than proportions of instances.

- **Post-pruning process**: C4.5 employs a technique known as *Pessimistic Error Pruning*. It computes, for a given sub-tree, an upper bound of the estimated error rate by means of a continuous correction of the Binomial distribution. The sub-tree hanging from a certain node is pruned if its upper bound is higher than the upper bound of the errors produced by the estimations of that node supposing that it acts as a terminal node.

4.6 Decision Trees based on imprecise probabilities

Credal Decision Tree (CDT) was introduced by Abellán and Moral [1]. The main difference between classical Decision Trees and CDT resides in the split criterion; while classical Decision Trees use classical probability theory, CDT employs imprecise probabilities.

Specifically, CDT uses the IDM to represent the information about the class variable from data at each node. Let $\mathcal{D}$ be the partition of the training set corresponding to a certain node and $N^{\mathcal{D}}$ the total number of instances in $\mathcal{D}$, i.e, $N^{\mathcal{D}} = |\mathcal{D}|$. For each $j = 1, 2, \ldots, K$, let $n^{\mathcal{D}}(c_j)$ denote the number of instances in $\mathcal{D}$ that satisfy $C = c_j$. CDT considers the IDM credal set on C corresponding to $\mathcal{D}$, determined via Equation (2.63):

$$\mathcal{P}^{\mathcal{D}}_{IDM}(C) = \left\{ p \in \mathcal{P}(C) \mid p(c_j) \in \left[\frac{n^{\mathcal{D}}(c_j)}{N^{\mathcal{D}} + s}, \frac{n^{\mathcal{D}}(c_j) + s}{N^{\mathcal{D}} + s} \right], \quad \forall j = 1, 2, \ldots, K \right\}, \tag{4.20}$$

s being the IDM parameter and $\mathcal{P}(C)$ the set of all probability distributions on C.

Uncertainty measures can be applied on the credal set $\mathcal{P}^{\mathcal{D}}_{IDM}(C)$. The procedure to build a CDT considers the maximum entropy on $\mathcal{P}^{\mathcal{D}}_{IDM}(C)$:

$$S^*(\mathcal{P}^{\mathcal{D}}_{IDM}(C)) = \max_{p \in \mathcal{P}^{\mathcal{D}}_{IDM}(C)} S(p), \tag{4.21}$$

where $S(p)$ is the Shannon entropy of the probability distribution p, determined by means of Equation (3.2).

As argued in Section 3.4, the maximum entropy is the well-established uncertainty measure on credal sets because it satisfies the required properties.

The split criterion employed in CDT is based on the *Imprecise Information Gain* (IIG) [1]. For each attribute X^i, the IIG value on $\mathcal{D}$ is computed as follows:

$$IIG^{\mathcal{D}}(C, X^i) = S^*(\mathcal{P}^{\mathcal{D}}_{IDM}(C)) - \sum_{r_i=1}^{t_i} \hat{p}^{\mathcal{D}}(x^i_{r_i}) S^*(\mathcal{P}^{\mathcal{D}}_{IDM}(C \mid X^i = x^i_{r_i})), \quad (4.22)$$

where $S^*(\mathcal{P}^{\mathcal{D}}_{IDM}(C \mid X = x^i_{r_i}))$ is the maximum entropy on the IDM credal set on C on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}$ and $\hat{p}^{\mathcal{D}}$ is the probability distribution that attains the maximum entropy on the IDM credal set on X^i corresponding to $\mathcal{D}$, $\mathcal{P}^{\mathcal{D}}_{IDM}(X^i)$, $\forall t_i = 1, 2, \dots, r_i$, $i = 1, 2, \dots, d$.

We may note that IIG differs from IG in that the latter criterion is based on the Shannon entropy on C while the former is based on the maximum entropy on the IDM credal set on C. Therefore, IIG uses the same idea as IG of measuring the gain in uncertainty-based information about the class variable. Nonetheless, whereas IG quantifies such information through precise probabilities, IIG employs the maximum entropy on credal sets for this purpose. It should be noted that, unlike IG, the value of IIG for an attribute can be negative [148].

4.6.1 IDM parameter in Credal Decision Trees

As commented in Section 2.3.1, Walley [209] recommends two values for the s parameter: $s = 1$ and $s = 2$. In addition, the procedure to compute $S^*(\mathcal{P}^{\mathcal{D}}_{IDM}(C))$ reaches its lowest computational cost when $s = 1$. Indeed, in Section 3.4.0.1, we have shown the procedure to obtain the probability distribution of maximum entropy on $\mathcal{P}^{\mathcal{D}}_{IDM}(C)$ for such a value. For these reasons, the value $s = 1$ is usually employed to build a CDT.

In Section 2.3.1, we have pointed out that IDM probability intervals are wider as the s value is higher. Thereby, in CDTs, the s value indicates the estimated imprecision degree in a certain node. Mantas, Abellán, and Castellano [150] showed both theoretically and experimentally that the s value should be higher as the level of class noise in the data is higher. It is because the over-fitting in a CDT decreases as the s value increases and vice-versa. In fact, each dataset has associated with it an optimal value of the IDM parameter in classification [18].

The original CDT method proposed by Abellán and Moral [1] uses the IDM for the split criterion. An experimental study carried out in [6] showed that the NPI-M and the A-NPI-M obtain statistically equivalent results to the IDM with the best choice of the s parameter when these models are used in CDTs for representing the uncertainty-based information about the class variable.

4.6.2 Credal C4.5

Within CDTs, a new version of the well-known C4.5 algorithm was proposed by Mantas and Abellán [149]. Such a version is called Credal C4.5 (CC4.5). The main difference between C4.5 and CC4.5 is that the former method quantifies the uncertainty-based information about the class variable via classical probability theory, while the latter uses the IDM for quantifying such information.

The basis of the split criterion of CC4.5 is IIG, defined in Equation (4.22). Similarly to C4.5, the split criterion of CC4.5 normalizes the IIG measure by dividing by the uncertainty-based information about the attribute. Hence, as C4.5, CC4.5 penalizes attributes with many values. For each attribute X^i, the split criterion of CC4.5, called *Imprecise Information Gain Ratio* (IIGR), is defined as follows:

$$\mathrm{IIGR}^{\mathcal{D}}(C, X^i) = \frac{\mathrm{IGR}^{\mathcal{D}}(C, X^i)}{S^*\left(\mathcal{P}^{\mathcal{D}}_{IDM}(X^i)\right)}, \tag{4.23}$$

where $\mathcal{P}^{\mathcal{D}}_{IDM}(X^i)$ is the IDM credal set on X^i associated with $\mathcal{D}$ and $S^*\left(\mathcal{P}^{\mathcal{D}}_{IDM}(X^i)\right)$ is the maximum entropy on $\mathcal{P}^{\mathcal{D}}_{IDM}(X^i)$, $\quad \forall i = 1, 2, \ldots, d$.

In this way, the main points of CC4.5 can be summarized as follows:

- **Split criterion**: At each node, CC4.5 selects the attribute the highest IIGR value whenever such a value is higher than the average IIGR value between the valid split attributes. As in C4.5, these valid split attributes are those whose number of possible values is lower than 0.3 times the number of instances in that branch or are numeric.

- **Stop branching criterion**: Similarly to C4.5, CC4.5 stops branching the tree when there is no attribute for which the IIGR value is positive, or there are no attributes that are numeric or whose number of possible values is lower than 0.3 times the number of instances in the corresponding branch. The stop branching criterion of CC4.5 also uses the criterion of minimum instances per leaf of C4.5.

- **Criterion for assigning class values to leaf nodes**: At each leaf node, as C4.5, CC4.5 assigns the most frequent class value in the subset of the training set corresponding to that terminal node (See Equation (4.19)).

- **Numeric attributes**: CC4.5 and C4.5 handle continuous attributes similarly. The only difference is that CC4.5 uses IIGR while C4.5 employs IGR.
- **Missing values**: The procedure of dealing with missing values of CC4.5 is similar to the one of C4.5. Again, the only difference is the use of IIGR by CC4.5 versus the use of IGR by C4.5.
- **Post-pruning process**: C4.5 and CC4.5 use the same post-pruning technique: the *Pessimistic Error Pruning*.

4.6.3 Credal Decision Trees versus classical Decision Trees

The main differences between the behavior of classical Decision Trees and CDTs can be summarized in the following points:

- **Size of the dataset**: It should be noted that, when $s = 0$, IDM credal sets only contain the probability distribution associated with relative frequencies and, consequently, classical Decision Trees are identical to CDTs. Nevertheless, when $s > 0$, IDM probability intervals are narrower as the size of the dataset is larger. Thus, at the upper levels of the tree, where there are often many instances, the values of IG and IIG might be similar as the probability distribution corresponding to relative frequencies is likely to be close to the one that attains the maximum entropy on the IDM credal set. In contrast, at the lower levels of the tree, where the number of instances tends to be small, the IDM credal set may contain many probability distributions quite different from the one estimated via relative frequencies and, therefore, the values of IG and IIG might not be close. In consequence, classical Decision Trees and CDTs might have a similar behavior at the upper levels of the tree but they probably behave very differently at the lower levels [148].
- **Negative values of the split criterion**: Unlike IG, the value of IIG can be negative for a certain attribute [149]. For this reason, CDTs avoid choosing attributes that worsen the uncertainty-based information about the class variable. Therefore, CDTs may stop branching the tree before classical Decision Trees and, thus, they probably over-fit less the data. It does not mean that CDTs under-fit the data since they select an attribute if it produces a gain of information about the class variable [149].
- **Robustness to noise**: Mantas, Abellán, and Castellano [150] showed that the maximum entropy on an IDM credal set is less sensitive to noise than

the classical Shannon entropy. Hence, as the main difference between classical Decision Trees and CDTs is the use of the mentioned measures, it can be stated that the former models are more sensitive to noise than the latter models. Indeed, experimental studies carried out in [148, 149] revealed that CDTs perform better than classical Decision Trees, the improvement being more significant as there is more class noise in the data.

Classical Decision Trees also handle noisy data through pruning processes. Indeed, such pruning processes can consider that the dataset in a certain node is not very reliable by removing the hanging sub-tree. Nevertheless, pruning processes can only deal with the noisy data by removing the sub-tree, whereas CDTs may be capable of handling such noisy data by adjusting the generated sub-tree in detail via uncertainty measures on credal sets for the split criterion [149].

4.7 Ensembles of classifiers

In many areas of science, such as finances or medicine, it is very common to take several opinions into account before making a decision. This idea has also been applied in classification by means of *ensembles*. Ensemble schemes learn multiple classifiers through individual classification algorithms. For classifying a new instance, predictions are made in the individual classifiers, and such predictions are combined to give a final prediction for that instance. The combination of the predictions tends to be made via a majority voting scheme.

The use of ensemble schemes might improve the results obtained by an individual classifier. Indeed, ensembles are usually more accurate and robust than individual classifiers [80]. For example, in credit scoring, the use of ensemble schemes has obtained better results than the use of individual classifiers. In consequence, if banks and financial institutions use ensembles instead of individual classifiers to make decisions about granting loans, then they might obtain considerable benefits [8, 151, 210].

We show below the main ensemble schemes developed so far for classification:

- **Bagging** [37]: This method, for each individual classifier, considers a bootstrap sample randomly drawn from the original training set with replacement. The size of each sample is equal to the size of the original training set. Hence, in each sample, some instances might appear more than once, while other instances may not appear. A classification model is learned from each one of these bootstrapped samples. Since the classifiers are built with different training sets, they are often different from each other. When classifying a new instance, the predictions made by the individual classifiers are combined via a majority vote.

- **Boosting** [184]: It also considers a bootstrap sample for each individual classifier. However, unlike Bagging, in Boosting, the re-sampling is directed to obtain the most informative data for each consecutive learner. When a new instance is required to be classified, the predictions of the individual classifiers, weighted by their accuracy, are combined.

  The most known Boosting method is Adaboost [91]. In such a method, the successive samples are obtained by re-weighting the training instances. Initially, the same weight is assigned to all instances. In each iteration, these weights are adjusted depending on the classification errors made by the individual classifier in such a way that instances erroneously classified are more likely to appear in the next sample.

- **Random Subspace** [114]: This approach, for each individual classifier, considers all training instances but only n_att features from the original attribute space, where n_att is a fixed parameter of the algorithm. Empirical studies have shown that a standard value of n_att that obtains good results is equal to half of the total number of attributes. In order to classify a new instance, the predictions are combined via the majority vote.

- **DECORATE (Diverse Ensemble Creation by Oppositional Relabelling of Artificial Training Examples)** [153]: It generates an ensemble by learning a new classifier in each iteration. The first classifier is built with the original training set. The remaining classifiers are built with an artificial training set resulting from the union of the original set and artificial instances, obtained by probabilistically estimating the value of each attribute from the data distribution [153]. The class values of the artificial instances are selected in such a way that they maximally differ from the current predictions to encourage diversity. For maintaining the training accuracy, the classifier is added to the ensemble scheme if, and only if, its incorporation does not decrease the performance of the ensemble. When a new instance is required to be classified, the predictions made by the classifiers of the ensemble are combined by means of a majority vote.

- **Rotation Forest** [180]: This ensemble approach, for each individual classifier, separates the features into N_F non-overlapping subsets equally sized and, as Bagging, randomly draws a bootstrap sample from the original training set with replacement. Then, a Principal Component Analysis[4] is run separately in each subset of features, and a new set of attributes is obtained to build the classifier. In order to classify a new instance, a majority vote is carried out on the predictions made by the individual classifiers.

- **Random Forest** [38]: This algorithm learns multiple Decision Trees. For each one of these trees, similarly to Bagging, Random Forest considers a bootstrap sample randomly drawn from the original training set with replacement. In each node of each tree of the ensemble, only n_att attributes are considered candidates for splitting the data. For classifying an instance, Random Forest combines the predictions made by the individual Decision Trees through a majority vote.

We may observe that, in Bagging, Adaboost, Random Subspace, DECORATE, and Rotation Forest, any classification algorithm can be employed to build the individual classifiers. In contrast, Random Forest only works with Decision Trees.

4.7.1 Diversity in ensembles

Breiman [37] argued that, for an ensemble to be successful, it is essential that the individual classifiers are not only accurate but also *diverse* or *unstable*. Actually, if the individual classifiers are very similar, then the performance of an ensemble of them might not be significantly better than the performance of any of these individual classifiers.

It is known that, in Decision Trees, small changes in the training set might produce considerable variations in the learned model. Hence, Decision Trees are very suitable for ensembles as they encourage diversity. In fact, in ensemble schemes, Decision Trees often achieve better results than more complex methods that perform better as individual classifiers [20, 151]. Also, we must remark that CDTs have supposed an improvement over classical Decision Trees when they are utilized in ensembles [8, 20, 22, 23].

Other ways for increasing diversity in an ensemble of classifiers are:

- Random choice of instances from the original training set with replacements to build each classifier. Examples of ensemble methods that use this procedure are Bagging, Random Forest, and Rotation forest. Some

[4] Principal Component Analysis is a procedure to obtain, for a dataset, a set of non-correlated attributes from the original attribute space of such a dataset. It is useful to reduce the dimensionality of the original dataset.

areas of the instance space may not be studied by individual classifiers because they are hidden by the more frequent instances. With the random selection of instances for each individual classifier, some of these classifiers can study these zones with less frequent instances and, thus, exploit some interesting characteristics for improving the accuracy and robustness of the ensemble method.

- Random selection of features from the original attribute set for each classifier. Examples of ensemble schemes with this property are Random Subspace and Random Forest. Some attributes might not be used by the individual classifiers because they are hidden by other more important attributes according to the criteria established by the algorithms. However, these hidden attributes can provide interesting information for the ensemble. In this way, the random selection of attributes can give an opportunity to the hidden attributes for providing their knowledge to the ensemble method. Thereby, the mentioned procedure can improve the performance of the ensemble.

4.8 Cost-sensitive classification

Standard classifiers aim to minimize the number of instances incorrectly classified, which is optimal when all classification errors have the same importance. In contrast, cost-sensitive classifiers takes the missclassification costs into account by attempting to minimize the total cost of instances incorrectly classified.

There are three main approaches for cost-sensitive classification [89]:

- **Direct approach**: It consists of directly considering the misclassification costs when training a classifier. For example, in [141], a Decision Tree whose split criterion directly takes the error costs into consideration was proposed.

- **Preprocessing the training data**: Cost-sensitive classification algorithms within this approach transform the training set by considering the misclassification costs. A well-known example is the MetaCost algorithm [81]. It is a wrapper method that uses an ensemble of classifiers with bootstrapped samples of the original training set. Then, it re-assigns the values of the class variable for the training instances so that the risk of the predictions made by the ensemble for them is minimized. Finally, a standard classifier is learned using this preprocessed training set. Another example is the Decision Tree proposed in [199], which considers weights for the training instances depending on their error costs.

- **Output adaptation**: The cost-sensitive classification methods belonging to this category utilize a standard classification algorithm for learning the model and, to classify a new instance, adapt the output by taking the error costs into account. An example of this approach is the adaptation of the Nearest Neighbors algorithm for cost-sensitive classification [108, 174].

In this work thesis, we only consider the Decision Tree that weights instances using the error costs, which we expose in Section 4.8.1.

4.8.1 Weighted Decision Tree

Let $\mathcal{M}$ be the matrix of errors costs of dimension $K \times K$, where the m_{ij} value indicates the cost of predicting, for an instance, the value c_i when the real class value is $c_j, \quad \forall i, j \in \{1, 2, \ldots, K\}$. It always holds that $m_{ii} = 0, \quad \forall i = 1, 2, \ldots, K$.

For each class value c_j, with $j = 1, 2, \ldots, K$, Weighted Decision Tree (Weighted-DT) [199] estimates the cost of incorrectly classifying an instance whose true class value is c_j. For this purpose, it employs the following conversion:

$$\mathrm{Cost}(j) = \sum_{i=1}^{K} m_{ij}, \quad \forall j = 1, 2, \ldots, K. \tag{4.24}$$

Using these costs, Weighted-DT computes weights for the training instances depending on their class values. Specifically, the weight of a training instance whose real class value is c_j is computed as follows:

$$w_j = \mathrm{Cost}(j) \times \frac{N_{tr}}{\sum_{i=1}^{K} n_{tr}(c_i) \times \mathrm{Cost}(i)}, \quad \forall j = 1, 2, \ldots, K, \tag{4.25}$$

where N_{tr} is the total number of instances in the training set and $n_{tr}(c_i)$ is the number of training instances for which $C = c_i, \quad \forall i = 1, 2, \ldots, K$. We may note that the sum of all instance weights is equal to $\sum_{j=1}^{K} w_j \times n_{tr}(c_j) = N_{tr}$.

Let $\mathcal{D}$ be the subset of the training set corresponding to a certain node. Let $n^{\mathcal{D}}(c_j)$ denote the number of instances in $\mathcal{D}$ for which $C = c_j$ and $W_j^{\mathcal{D}}$ the sum of weights of such instances:

$$W_j^{\mathcal{D}} = n^{\mathcal{D}}(c_j) \times w_j, \quad \forall j = 1, 2, \ldots, K. \tag{4.26}$$

Let $W^{\mathcal{D}}$ be the total sum of weights in the node:

$$W^{\mathcal{D}} = \sum_{j=1}^{K} n^{\mathcal{D}}(c_j) \times w_j. \tag{4.27}$$

Weighted-DT estimates the probability of each class value in the node as follows:

$$p^{\mathcal{D}}(c_j) = \frac{W_j^{\mathcal{D}}}{W^{\mathcal{D}}}, \quad \forall j = 1, 2, \dots, K. \tag{4.28}$$

In this way, for the estimation of the probability distribution of C on $\mathcal{D}$, an instance has more importance as the misclassification cost of its corresponding class value is higher.

The basis of the split criterion of Weighted-DT is the Shannon entropy of $p^{\mathcal{D}}$:

$$S^{\mathcal{D}}(C) = -\sum_{j=1}^{K} p^{\mathcal{D}}(c_j) \log_2(p^{\mathcal{D}}(c_j)). \tag{4.29}$$

Remark that the Shannon entropy is the well-established uncertainty measure for probability distributions.

The split criterion of Weighted-DT is called *Weighted Information Gain* (WIG). It is defined, for an attribute X^i whose set of possible values is $\{x_1^i, x_2^i, \dots, x_{t_i}^i\}$, as follows:

$$WIG(C, X^i) = S^{\mathcal{D}}(C) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x_{r_i}^i) S^{\mathcal{D}}(C \mid X^i = x_{r_i}^i), \tag{4.30}$$

where $S^{\mathcal{D}}(C \mid X^i = x_{r_i}^i)$ is the Shannon entropy on C on the partition of $\mathcal{D}$ composed of those instances for which $X^i = x_{r_i}^i$, computed similarly to $S^{\mathcal{D}}(C)$, and $P^{\mathcal{D}}(X^i = x_{r_i}^i)$ is the probability that $X^i = x_{r_i}^i$ on $\mathcal{D}$, estimated via proportion of weights, $\forall r_i = 1, 2, \dots, t_i, \quad i = 1, 2, \dots, d$.

For classifying an instance at a leaf node, Weighted-DT predicts the class value with the highest sum of weights. Formally, let $n^{\mathcal{L}}(c_j)$ denote the number of instances at a leaf node $\mathcal{L}$ for which $C = c_j, \quad \forall j = 1, 2, \dots, K$. Weighted-DT predicts at $\mathcal{L}$ the class value c_k that satisfies:

$$k = \arg \max_{j=1,2,\dots,K} n^{\mathcal{L}}(c_j) \times w_j. \tag{4.31}$$

Consequently, at each terminal node, the instances whose class value has a higher misclassification cost have more importance.

5 IMPRECISE CLASSIFICATION

5.1 Introduction

In order to classify an instance, classifiers often predict a single value of the class variable. Nevertheless, in many cases, there is not enough information available for classifiers to point out a unique class value. In these cases, it makes more sense that classifiers predict a set of class values. This type of prediction is called *imprecise prediction*, and classifiers that make imprecise predictions are called *imprecise classifiers* [227].

When an imprecise classifier is employed, a set of values of the class variable may be obtained, known as the *non-dominated states set*. It is composed of those class values that are not “defeated" by another one according to an established criterion, usually called the *dominance criterion*. Several dominance criteria have been proposed so far for Imprecise Classification. A comparative study of them can be found in [17].

An evaluation metric for an imprecise classifier has to consider whether the predictions are correct, i.e, whether the real class values belong to the non-dominated states sets and how informative the predictions are, which is measured by the cardinalities of the predicted sets of class values. If an imprecise classifier also takes the error costs into account, then an evaluation metric must consider the costs of instances incorrectly classified and the number of predicted class values (how informative the predictions are).

Few Imprecise Classification methods have been developed so far. All of them use imprecise probability models because imprecise probabilities are more appropriate than classical probability theory for Imprecise Classification algorithms [10]. The first Imprecise Classification method was the *Naïve Credal Classifier* (NCC) [62, 227]. It combines the IDM with the naïve assumption to make imprecise predictions. Afterwards, Abellán and Masegosa [10] proposed the first Imprecise Classification algorithm based on Decision Trees. It uses the same tree-building process as the Credal Decision Tree algorithm and utilizes a dominance criterion at leaf nodes for making imprecise predictions. The mentioned algorithms were also adapted for considering error costs by Abellán and Masegosa [10].

The remainder of this chapter is structured as follows: Section 5.2 explains the Imprecise Classification paradigm. The main evaluation metrics proposed so far for Imprecise Classification are exposed in Section 5.3. In Section 5.4, we describe the main dominance criteria proposed so far for Imprecise Classification. Section 5.5 details the Naïve Credal Classifier. The Imprecise Classification algorithm based on Decision Trees is described in Section 5.6.

5.2 Imprecise Classification problem

As traditional classification, the Imprecise Classification task starts from the following issues:

- A set of d predictive attributes $\{X^1, X^2, \dots, X^d\}$. Let $Dom(X^i)$ denote the domain of the X^i attribute, $\quad \forall i = 1, 2, \dots, d$.
- A class variable C, whose set of possible values is $\Omega_C = \{c_1, c_2, \dots, c_K\}$. Let 2^{Ω_C} denote the power set of Ω_C.

Imprecise Classification aims to learn a model $h : \left(Dom(X^1), Dom(X^2), \dots, Dom(X^d)\right) \rightarrow 2^{\Omega_C}$ which, for a new instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}\right)$, where $x^i_{r_i} \in Dom(X^i) \quad \forall i = 1, 2, \dots, d$, returns the predicted non-dominated states set for that instance, namely $h(\mathbf{x})$.

Similarly to traditional classification, for learning the Imprecise Classification model h, a training set $\mathcal{D}_{train}$ is often utilized, where each instance is described via a set of attribute values and has a unique value of the class variable.

Within this section, we assume that each attribute X^i takes values in a finite set, that is, $Dom(X^i) = \left\{x^i_1, x^2_2, \dots, x^i_{t_i}\right\}, \quad \forall i = 1, 2, \dots, d$.

5.3 Evaluation metrics for Imprecise Classification

In order to evaluate the performance of an imprecise classifier described by a function $h : \left(Dom(X^1), Dom(X^2), \dots, Dom(X^d)\right) \rightarrow 2^{\Omega_C}$, a test set $\mathcal{D}_{test}$ is frequently used, as in traditional classification. Let $x^i_{r_{ij}}$ denote the value of the ith attribute for the jth test instance, where $r_{ij} \in \{1, 2, \dots, t_i\}$, $\mathbf{x_j} = \left(x^1_{r_{1j}}, x^2_{r_{2j}}, \dots, x^d_{r_{dj}}\right)$, and $c^j \in \Omega_C$ the class value of the jth test instance, $\quad \forall i = 1, 2, \dots, d, \quad j = 1, 2, \dots, N_{test}$.

As commented before, an evaluation metric for an imprecise classifier must take into account two issues: Accuracy (whether the real class value belongs to the non-dominated states set) and informativeness (number of predicted class values).

The following two metrics, proposed by Corani and Zaffalon [62], are useful to evaluate how informative the predictions are:

- **Determinacy**: The proportion of test instances for which a single class value is predicted:

$$\text{Determinacy}(h) = \frac{1}{N_{test}} \sum_{j=1}^{N_{test}} \left[\left[\left|h(\mathbf{x_j})\right| = 1\right]\right]. \tag{5.1}$$

- **Indeterminacy size**: The average number of predicted class values between the test instances with two or more non-dominated states:

$$\text{Indeterminacy\_size}(h) = \frac{1}{N_{impr}} \sum_{j=1,\left|h(\mathbf{x_j})\right|>1}^{N_{test}} \left|h(\mathbf{x_j})\right|, \tag{5.2}$$

  where $N_{impr} = \left|\left\{j \in \{1,2,\ldots,N_{test}\} : \left|h(\mathbf{x_j})\right| > 1\right\}\right|$.

Regarding correct predictions, Corani and Zaffalon [62] introduced the following evaluation measures:

- **Single Accuracy**: The accuracy between the test instances precisely classified:

$$\text{Single\_Accuracy}(h) = \frac{1}{N_{prec}} \sum_{j=1,\left|h(\mathbf{x_j})\right|=1}^{N_{test}} \left[\left[h(\mathbf{x_j}) = \{c^j\}\right]\right], \tag{5.3}$$

  where $N_{prec} = \left|\left\{j \in \{1,2,\ldots,N_{test}\} : \left|h(\mathbf{x_j})\right| = 1\right\}\right|$.

- **Set Accuracy**: It indicates, between the test instances indeterminately classified, the proportion of them for which the real class value belongs to the predicted non-dominated states set:

$$\text{Set\_Accuracy}(h) = \frac{1}{N_{impr}} \sum_{j=1,\left|h(\mathbf{x_j})\right|>1}^{N_{test}} \left[\left[c^j \in h(\mathbf{x_j})\right]\right]. \tag{5.4}$$

For evaluating the whole performance of an imprecise classifier, that is, its trade-off between Accuracy and informative predictions, Corani and Zaffalon [62] proposed the *Discounted Accuracy* metric (DACC). It is defined in the following way:

$$\mathrm{DACC}(h) = \frac{1}{N_{test}} \sum_{j=1}^{N_{test}} \frac{[[c^j \in h(\mathbf{x_j})]]}{|h(\mathbf{x_j})|}. \tag{5.5}$$

We may note that DACC is an accuracy measure that penalizes the right predictions by dividing by the number of predicted class values. It reaches its maximum value, which is equal to 1, when all predictions are correct and precise. The minimum value of DACC, 0, is attained when all predictions are incorrect. If an imprecise classifier always predicts all class values, then the DACC value is equal to $\frac{1}{K}$. This might be a drawback of the DACC measure since, in this situation, the classifier is not informative.

5.3.1 Evaluating cost-sensitive imprecise classifiers

The DACC measure does not penalize errors in a strict sense as it does not add any negative value when an instance is misclassified; it is only an accuracy measure. Thereby, DACC is not sufficient for checking the performance of an imprecise classifier when different classification errors yield different costs.

In order to solve this issue, Abellán and Masegosa [10] proposed a new evaluation metric called the *Measure for Imprecise Classifiers* (MIC). Such an evaluation metric considers two points:

- If the prediction made for the jth test instance is correct, then a positive value has to be added, which must be inversely proportional to the number of predicted non-dominated states. Hence, in this case, MIC adds the value $-\log_2\left(\frac{|h(\mathbf{x_j})|}{K}\right)$.
- When the real class value of the jth test instance does not belong to the non-dominated states set predicted for such an instance, the maximum cost of predicting a class value belonging to the non-dominated states set is considered. As pointed out in [10], it makes sense to take the maximum cost because, in practical applications, an user sometimes has to choose a class value among the predicted ones. In this way, when the prediction made for the jth test instance is not correct, MIC adds the negative value $-\alpha_j \log_2(K)$, where:

$$\alpha_j = \max_{c_k \in h(\mathbf{x_j})} m(c_k, c^j), \tag{5.6}$$

$m(c_k, c^j)$ being the cost of predicting the c_k value when the real class value is c^j, $\forall k = 1, 2, \dots, K, \quad j = 1, 2, \dots, N_{test}$.

Then, MIC is defined in the following way:

$$\mathrm{MIC}(h) = \sum_{j=1, c^j \in h(\mathbf{x}_j)}^{N_{test}} -\log_2\left(\frac{|h(\mathbf{x}_j)|}{K}\right) - \frac{1}{K-1} \sum_{j=1, c^j \notin h(\mathbf{x}_j)}^{N_{test}} \alpha_j \log_2 K. \quad (5.7)$$

We may observe that the maximum value of MIC, which is equal to $\log_2(K)$, is attained when all test instances are correctly and precisely classified. In addition, if $|h(\mathbf{x}_j)| = K$, then $\log_2\left(\frac{|h(\mathbf{x}_j)|}{K}\right) = 0$. Thus, when an imprecise classifier always predicts all class values, the MIC value is equal to 0. It makes sense as, in such a situation, the classifier is not informative.

MIC is also useful when all classification errors have the same cost (1) [10]. In this case, MIC is determined in the following way:

$$\mathrm{MIC}^{0/1}(h) = -\sum_{j=1, c^j \in h(\mathbf{x}_j)}^{N_{test}} -\log_2\left(\frac{|h(\mathbf{x}_j)|}{K}\right) - \frac{1}{K-1} \sum_{j=1, c^j \notin h(\mathbf{x}_j)}^{N_{test}} \log_2 K. \quad (5.8)$$

As in traditional classification, a cross-validation procedure tends to be used to estimate the performance of an imprecise classifier via the evaluation metrics explained above. Moreover, the same statistical tests are employed for comparing the results obtained by two or more imprecise classifiers in the cross-validation procedure.

5.4 Dominance criteria in Imprecise Classification

For classifying an instance, an Imprecise Classification algorithm needs to select one or more alternatives among the possible values of the class variable. In order to make such a decision, it must use the probabilistic knowledge about the class variable for the instance to classify, which is usually determined by means of an imprecise probability model. Several works have been carried out in the literature concerning *decision-making* with imprecise probabilities [105, 162, 163, 200].

In classification, the decision-making process with imprecise probabilities can be made by directly using the lower and upper probabilities of the class values [17]. Suppose that the probabilistic knowledge about the class variable C for a given instance is determined via a credal set on C, namely $\mathcal{P}^C$. The first dominance criterion, called the *stochastic dominance*, was introduced by Luce and Raiffa [143]. That criterion establishes that a class value dominates another one if the lower probability of the former class value is greater than the upper probability of the latter class value. Formally:

Definition 5.4.1 *Let $\underline{P}\left(\mathcal{P}^C\right)$ and $\overline{P}\left(\mathcal{P}^C\right)$ denote, respectively, the coherent lower and upper probability functions derived from $\mathcal{P}^C$, computed through Equation (2.15). It is said that there is stochastic dominance of c_j on c_k under $\mathcal{P}^C$ if*

$$\underline{P}\left(\mathcal{P}^C\right)\left(\{c_j\}\right) > \overline{P}\left(\mathcal{P}^C\right)(\{c_k\}), \quad \forall j,k \in \{1,2,\ldots,K\}.$$

It should be noted that stochastic dominance is a very strict dominance criterion. Zaffalon [227] showed that it is possible that the probability of a class value is always greater than the probability of another class value even though the upper probability of the latter class value is greater than the lower probability of the former. In these cases, it makes sense that the former class value dominates the latter. For this reason, Zaffalon [227] defined the *credal dominance* criterion, according to which a class value dominates another one if, and only if, for all probability distribution on $\mathcal{P}^C$, the probability of the former class value is greater than the probability of the latter. Formally:

Definition 5.4.2 *It is said that there is credal dominance of c_j on c_k under $\mathcal{P}^C$ if* $p(c_j) > p(c_k) \quad \forall p \in \mathcal{P}^C$.

According to the results proved by Abellán [17], stochastic dominance always implies credal dominance, but the converse is not true. In consequence, credal dominance is a more informative criterion than stochastic dominance. However, the latter criterion is normally far softer to check than the former and, thus, it is more practical [17].

5.4.1 Dominance criterion on reachable probability intervals

Let us assume now that probabilistic knowledge about C is represented by a reachable set of probability intervals on C, namely
$\mathcal{I}(C) = \left\{\left[l_j, u_j\right], \quad j = 1,2,\ldots,K\right\}$. Let $\mathcal{P}\left(\mathcal{I}(C)\right)$ denote the credal set associated with $\mathcal{I}(C)$, computed by means of Equation (2.47). The following result, demonstrated by Abellán [17], highlights that, in this case, stochastic and credal dominance are equivalent.

Proposition 5.4.1 *For each* $\{j,k\} \subseteq \{1,2,\ldots,K\}$, *it holds that* $l_j > u_k \Leftrightarrow p(c_j) > p(c_k) \quad \forall p \in \mathcal{P}\left(\mathcal{I}(C)\right)$.

Therefore, when the probabilistic knowledge about C is represented via a reachable set of probability intervals on C, to obtain the cases of credal dominance (the most informative dominance criterion), it is just necessary to consider the bounds of the intervals for the class values.

5.4.2 Dominance criterion for cost-sensitive imprecise classifiers

So far, the only algorithms for cost-sensitive Imprecise Classification, introduced by Abellán and Masegosa [10], compute the non-dominated states set for an instance via a dominance criterion on the risk intervals on the class values for that instance.

Let $\mathcal{R} = \left\{ \left[\underline{R}(c_j), \overline{R}(c_j) \right], \quad j = 1, 2, \dots, K \right\}$ be the set of risk intervals on the class variable for an instance. In the algorithms for cost-sensitive Imprecise Classification proposed in [10], the *stochastic dominance* criterion is applied to $\mathcal{R}$ for obtaining the non-dominated states set.

Definition 5.4.3 *It is said that there is stochastic dominance of* c_j *on* c_k *under* $\mathcal{R}$ *if* $\overline{R}(c_j) < \underline{R}(c_k), \quad \forall j, k \in \{1, 2, \dots, K\}$.

We may note that this concept is quite intuitive and is based on the concept of stochastic dominance for probability intervals. Since we are working on risk intervals and not on credal sets, the credal dominance criterion does not make sense here [10].

5.5 The Naive Credal Classifier

The basis of the Naïve Credal Classifier (NCC) [62, 227] is the naïve assumption (given the class variable, all attributes are independent):

$$\begin{aligned}
&P\left(C = c_j \mid X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}\right) = \\
&\frac{P\left(C = c_j, X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}\right)}{P\left(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}\right)} \sim \\
&P\left(C = c_j, X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}\right) = \\
&P\left(C = c_j\right) P\left(X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d} \mid C = c_j\right) = \\
&P\left(C = c_j\right) \prod_{i=1}^{d} P\left(X^i = x^i_{r_i} \mid C = c_j\right), \\
&\forall j = 1, 2, \dots, K, \quad r_i = 1, 2 \dots, t_i, \quad i = 1, 2, \dots, d.
\end{aligned} \tag{5.9}$$

The proportionality relation indicated via the symbol $\sim$ is because $\arg\max_{j=1,2,\ldots,K}\left(\frac{P\left(C=c_j,X^1=x^1_{r_1},X^2=x^2_{r_2},\ldots,X^d=x^d_{r_d}\right)}{P\left(X^1=x^1_{r_1},X^2=x^2_{r_2},\ldots,X^d=x^d_{r_d}\right)}\right) =$ $\arg\max_{j=1,2,\ldots,K}\left(P\left(C=c_j,X^1=x^1_{r_1},X^2=x^2_{r_2},\ldots,X^d=x^d_{r_d}\right)\right)$.

Let $\mathcal{P}_{NCC}(C)$ be a credal set on the class variable C. For each $i = 1,2,\ldots,d$, $k = 1,2,\ldots,K$, let $\mathcal{P}_{NCC}\left(X^i \mid c_k\right)$ be a credal set on the attribute X^i conditioned on $C = c_k$.

Definition 5.5.1 *[227] The aforementioned credal sets are called local credal sets.*

The NCC algorithm considers, for each $k = 1,2,\ldots,K$, the set of joint probability distributions $\mathcal{P}_{NCC}\left(c_k,X^1,X^2,\ldots,X^d\right)$ obtained from the naïve assumption and making every possible combination of probability distributions on the local credal sets:

$$\begin{aligned}\mathcal{P}_{NCC}\left(c_k,X^1,X^2,\ldots,X^d\right) = \Bigg\{&p_c(c_k)\prod_{i=1}^{d}p_{ik}\\ &\mid p_c \in \mathcal{P}_{NCC}(C),\, p_{ik} \in \mathcal{P}_{NCC}\left(X^i \mid c_k\right)\Bigg\}.\end{aligned} \tag{5.10}$$

Hence, in order to build the NCC, only the local credal sets are required.

Let N_{tr} be the number of training instances. Let $n_{tr}(c_j)$ denote the number of training instances that satisfy $C = c_j$, $n_{tr}(x^i_{r_i})$ the number of training instances for which $X^i = x^i_{r_i}$, and $n_{tr}(x^i_{r_i}, c_j)$ the number of training instances that verify $C = c_j \wedge X^i = r_i$, $\forall j = 1,2,\ldots,K$, $r_i = 1,2,\ldots,t_i$, $i = 1,2,\ldots,d$.

For obtaining the local credal sets, the IDM is usually employed in the literature. We have the following set of IDM probability intervals on C:

$$\mathcal{I}_{IDM}(C) = \left\{I_{IDM}(c_k) = \left[\frac{n_{tr}(c_k)}{N_{tr}+s}, \frac{n_{tr}(c_k)+s}{N_{tr}+s}\right], \quad k = 1,2,\ldots,K\right\}. \tag{5.11}$$

In NCC, the local credal set on C is the credal set consistent with the intervals given in Equation (5.11):

$$\mathcal{P}_{NCC}(C) = \{p \in \mathcal{P}(C) \mid p(c_k) \in I_{IDM}(c_k), \quad \forall k = 1,2,\ldots,K\}, \tag{5.12}$$

$\mathcal{P}(C)$ being the set of all probability distributions on C.

Likewise, for $\mathcal{P}_{NCC}\left(X^i \mid c_k\right)$, NCC considers the credal set associated with the IDM probability intervals on X^i conditioned on $C = c_k$:

$$\mathcal{P}_{NCC}\left(X^i \mid c_k\right) = \left\{p \in \mathcal{P}(X^i \mid c_k) \mid p(x^i_{r_i} \mid c_k) \in I_{IDM}(x^i_{r_i} \mid c_k), \forall r_i = 1,\ldots,t_i\right\}, \tag{5.13}$$

where $I_{IDM}(x^i_{r_i} \mid c_k) = \left[\frac{n_{tr}(x^i_{r_i},c_j)}{n_{tr}(c_k)+s}, \frac{n_{tr}(x^i_{r_i},c_j)+s}{n_{tr}(c_k)+s}\right]$ and $\mathcal{P}(X^i \mid c_k)$ is the set of all probability distributions on X^i conditioned on $C = c_k$, $\forall r_i = 1,2,\dots,t_i$, $i = 1,2,\dots,d$, $k = 1,2,\dots,K$.

For obtaining the non-dominated states set for an instance such that $X^i = x^i_{r_i}$, with $r_i \in \{1,2,\dots,t_i\}$ $\forall i = 1,2,\dots,d$, NCC considers, for each class value, the lower and upper probabilities on the set of joint probability distributions determined by Equation (5.10).

We shall denote, for each $k = 1,2,\dots,K$, $r_i = 1,2,\dots,t_i$, $i = 1,2,\dots,d$:

$$\begin{aligned}
&\underline{p}(c_k) = \min_{p \in \mathcal{P}_{NCC}(C)} p(c_k), \quad \overline{p}(c_k) = \max_{p \in \mathcal{P}_{NCC}(C)} p(c_k),\\
&\underline{p}(x^i_{r_i} \mid c_k) = \min_{p_{ik} \in \mathcal{P}_{NCC}(X^i|c_k)} p_{ik}(x^i_{r_i} \mid c_k),\\
&\overline{p}(x^i_{r_i} \mid c_k) = \max_{p_{ik} \in \mathcal{P}_{NCC}(X^i|c_k)} p_{ik}(x^i_{r_i} \mid c_k).
\end{aligned} \tag{5.14}$$

It is easy to deduce that, $\forall k = 1,2,\dots,K$:

$$\begin{aligned}
\min_{p_c \in \mathcal{P}_{NCC}(C), p_{ik} \in \mathcal{P}_{NCC}(X^i|c_k)} \left\{ p_c(c_k) \prod_{i=1}^{d} p_{ik}(x^i_{r_i} \mid c_k) \right\} &= \underline{p}(c_k) \prod_{i=1}^{d} \underline{p}(x^i_{r_i} \mid c_k),\\
\max_{p_c \in \mathcal{P}_{NCC}(C), p_{ik} \in \mathcal{P}_{NCC}(X^i|c_k)} \left\{ p_c(c_k) \prod_{i=1}^{d} p_{ik}(x^i_{r_i} \mid c_k) \right\} &= \overline{p}(c_k) \prod_{i=1}^{d} \overline{p}(x^i_{r_i} \mid c_k).
\end{aligned}$$

Since IDM probability intervals are always reachable, under this model, the stochastic and credal dominance criteria are equivalent. In this way, under NCC, c_k dominates c_j if, and only if,

$$\underline{p}(c_k) \prod_{i=1}^{d} \underline{p}(x^i_{r_i} \mid c_k) \geqslant \overline{p}(c_j) \prod_{i=1}^{d} \overline{p}(x^i_{r_i} \mid c_j), \quad \forall j,k = 1,2,\dots,K.$$

Under the IDM:

$$\begin{aligned}
&\underline{p}(c_k) = \frac{n_{tr}(c_k)}{N_{tr}+s}, \quad \overline{p}(c_k) = \frac{n_{tr}(c_k)+s}{N_{tr}+s},\\
&\underline{p}(x^i_{r_i} \mid c_k) = \frac{n_{tr}(x^i_{r_i},c_k)}{n_{tr}(c_k)+s}, \quad \overline{p}(x^i_{r_i} \mid c_k) = \frac{n_{tr}(x^i_{r_i},c_k)+s}{n_{tr}(c_k)+s},\\
&\forall k = 1,2,\dots,K.
\end{aligned} \tag{5.15}$$

Consequently, in NCC, c_k dominates c_j if, and only if:

$$\frac{n_{tr}(c_k)}{N_{tr}+s}\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_k)}{n_{tr}(c_k)+s} \geqslant \frac{n_{tr}(c_j)+s}{N_{tr}+s}\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_j)+s}{n_{tr}(c_j)+s} \Leftrightarrow$$
$$n_{tr}(c_k)\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_k)}{n_{tr}(c_k)+s} \geqslant \left(n_{tr}(c_j)+s\right)\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_j)+s}{n_{tr}(c_j)+s},$$
$$\forall j,k = 1,2,\dots,K.$$

Therefore, given an instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}\right)$, with $r_i \in \{1,2,\dots,t_i\} \quad \forall i = 1,2,\dots,d$, the non-dominated states set predicted by NCC is determined in the following way:

$$h^{NCC}(\mathbf{x}) = \Bigg\{ c_j \mid (n_{tr}(c_j)+s)\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_j)+s}{n_{tr}(c_j)+s} > n_{tr}(c_k)\prod_{i=1}^{d}\frac{n_{tr}(x^i_{r_i},c_k)}{n_{tr}(c_k)+s}, \quad \forall k = 1,2,\dots,K \Bigg\}. \tag{5.16}$$

5.5.1 Adaptation for cost-sensitive scenarios

NCC was also adapted for cost-sensitive classification by Abellán and Masegosa [10]. Such an adaptation, called the cost-sensitive Naive Credal Classifier (CS-NCC), to classify an instance, computes the lower and upper probabilities as NCC and obtains a risk interval for each class value from these lower and upper probabilities and the error costs. Finally, it applies the stochastic dominance criterion on these risk intervals to obtain the non-dominated states set for the instance to classify.

Formally, suppose that it is required to classify an instance for which $X^i = x^i_{r_i}$, with $r_i \in \{1,2,\dots,t_i\} \quad \forall i = 1,2,\dots,d$. Let $\underline{P}_{NCC}(c_j)$ and $\overline{P}_{NCC}(c_j)$ denote, respectively, the lower and upper probabilities estimated by NCC for the c_j value:

$$\underline{P}_{NCC}(c_j) = \underline{p}(c_j)\prod_{i=1}^{d}\underline{p}(x^i_{r_i} \mid c_j), \quad \overline{P}_{NCC}(c_j) = \overline{p}(c_j)\prod_{i=1}^{d}\overline{p}(x^i_{r_i} \mid c_j), \tag{5.17}$$

where $\underline{p}(c_j)$, $\overline{p}(c_j)$, $\underline{p}(x^i_{r_i} \mid c_j)$, and $\overline{p}(x^i_{r_i} \mid c_j)$ are computed through Equation (5.15), $\quad \forall j = 1,2,\dots,K, \quad i = 1,2,\dots,d$.

Let $\mathcal{M}$ be the matrix of error costs defined in Section 4.8.1. From the probability intervals determined via Equation (5.17), CS-NCC computes a risk interval for each class value, where the lower (upper) risk is computed by considering the costs of predicting that class value when the real class value is another one and the lower (upper) probabilities of the remaining class values:

$$\underline{R}_{NCC}(c_j) = \sum_{k=1}^{K} m_{jk}\underline{P}_{NCC}(c_k), \qquad \overline{R}_{NCC}(c_j) = \sum_{k=11}^{K} m_{jk}\overline{P}_{NCC}(c_k), \quad \forall j = 1, 2, \dots, K. \tag{5.18}$$

CS-NCC determines the non-dominated states set for the instance via the stochastic dominance criterion on the risk intervals given by Equation (5.18), according to which a class value c_j dominates another one c_k if, only if, $\overline{R}_{NCC}(c_j) < \underline{R}_{NCC}(c_k), \quad \forall j, k \in \{1, 2, \dots, K\}$.

Thus, the non-dominated states set predicted by CS-NCC for an instance with attribute vector $\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d})$, where $r_i \in \{1, 2, \dots, t_i\} \quad \forall i = 1, 2, \dots, d$, is determined by:

$$h^{CS\_NCC}(\mathbf{x}) = \left\{c_j \mid \underline{R}_{NCC}(c_j) \geqslant \overline{R}_{NCC}(c_k), \quad \forall j = 1, 2, \dots, K\right\}, \tag{5.19}$$

where $\underline{R}_{NCC}(c_j)$ and $\overline{R}_{NCC}(c_j)$ are computed by means of Equation (5.18), $\forall j = 1, 2, \dots, K$.

5.6 Imprecise Credal Decision Tree

The Imprecise Credal Decision Tree method (ICDT), developed by Abellán and Masegosa [10], is an adaptation of the CDT algorithm for Imprecise Classification. Both methods use the same tree-building process.

Hence, at each node, the attribute with the highest Imprecise Information Gain value, computed by means of Equation (4.22), is selected.

CDT and ICDT differ in the procedure to classify instances at leaf nodes: while CDT predicts the most frequent class value at a leaf node, ICDT computes a probability interval for each possible value of the class variable and then applies a dominance criterion on these intervals to obtain the non-dominated states set. Formally, at a leaf node $\mathcal{L}$, let $n^{\mathcal{L}}(c_j)$ denote the number of instances in $\mathcal{L}$ that satisfy $C = c_j, \quad \forall j = 1, 2, \dots, K$, and $N^{\mathcal{L}}$ the total number of instances in $\mathcal{L}$. ICDT considers the set of IDM probability intervals on C on $\mathcal{L}$:

$$\mathcal{I}^{\mathcal{L}}_{IDM}(C) = \left\{\left[\frac{n^{\mathcal{L}}(c_j)}{N^{\mathcal{L}}+s}, \frac{n^{\mathcal{L}}(c_j)+s}{N^{\mathcal{L}}+s}\right], \quad j = 1, 2, \dots, K\right\}. \tag{5.20}$$

As argued in Section 5.4, the most informative dominance criterion is credal dominance and, as IDM probability intervals are always reachable, under this model, the stochastic and credal dominance criteria are equivalent. Thereby, in ICDT, a class value c_j dominates another one c_k at $\mathcal{L}$ if, and only if,

$$\frac{n^{\mathcal{L}}(c_j)}{N^{\mathcal{L}}+s} > \frac{n^{\mathcal{L}}(c_k)+s}{N^{\mathcal{L}}+s} \Leftrightarrow n^{\mathcal{L}}(c_j) > n^{\mathcal{L}}(c_k)+s, \quad \forall j,k \in \{1,2,\ldots,K\}.$$

In consequence, at the leaf node $\mathcal{L}$, the non-dominated states set predicted by ICDT is determined as follows:

$$nds^{\mathcal{L}}_{ICDT} = \left\{c_k \mid n^{\mathcal{L}}(c_k)+s \geqslant n^{\mathcal{L}}(c_j), \quad \forall j = 1,2,\ldots,K\right\}. \tag{5.21}$$

Similar to Decision Trees for precise classification, for classifying an instance via ICDT, a path from the root node to a leaf one is made by using the attribute values of that instance. Then, the stochastic dominance is applied to the probability intervals at that leaf node to obtain the non-dominated states set for the instance. Algorithm 7 summarizes the procedure to classify an instance with ICDT.

Algorithm 7: Procedure to classify an instance with ICDT.

```
Procedure Classify_ICDT(ICDT $\mathcal{T}$, instance with attribute vector
$\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d})$)
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the
   attribute values $x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}$.
2. Consider the set of IDM probability intervals on C at $\mathcal{L}$, $\mathcal{I}^{\mathcal{L}}_{IDM}(C)$,
   computed through Equation (5.20).
3. Obtain the non-dominated states set at $\mathcal{L}$ via the stochastic
   dominance criterion on $\mathcal{I}^{\mathcal{L}}_{IDM}(C)$:

                 $h^{ICDT}(\mathbf{x}) = nds^{\mathcal{L}}_{ICDT}$,

   where $nds^{\mathcal{L}}_{ICDT}$ is determined via Equation (5.21).
return $h^{ICDT}(\mathbf{x})$
```

5.6.1 Adaptation for cost-sensitive classification

The ICDT algorithm was also adapted for cost-sensitive classification by Abellán and Masegosa [10]. We call such an adaptation the cost-sensitive Imprecise Credal Decision Tree (CS-ICDT). CDT, ICDT, and CS-ICDT employ the same procedure to build the tree.

CS-ICDT differs from ICDT in the criterion utilized to classify instances at leaf nodes: ICDT uses the stochastic dominance criterion on the IDM probability intervals on C at a leaf node, whereas CS-ICDT considers the risk intervals on the class values at that leaf node.

Formally, let $\mathcal{M}$ be the matrix of error costs defined in Section 4.8.1. Let $\underline{P}^{\mathcal{L}}_{IDM}(c_j)$ and $\overline{P}^{\mathcal{L}}_{IDM}(c_j)$ denote, respectively the IDM lower and upper probabilities for the c_j value at $\mathcal{L}$:

$$\underline{P}^{\mathcal{L}}_{IDM}(c_j) = \frac{n^{\mathcal{L}}(c_j)}{N^{\mathcal{L}} + s}, \quad \overline{P}^{\mathcal{L}}_{IDM}(c_j) = \frac{n^{\mathcal{L}}(c_j) + s}{N^{\mathcal{L}} + s}, \quad \forall j = 1, 2, \dots, K. \tag{5.22}$$

From these probability intervals, a risk interval is computed for each class value, where the lower (upper risk) is computed by considering the costs of predicting that class value when the real class value is another one and the lower (upper) probabilities of the remaining class values:

$$\begin{aligned} \underline{R}_{CS-ICDT}(c_j) &= \sum_{k=1}^{K} m_{jk}\underline{P}^{\mathcal{L}}_{IDM}(c_k), \\ \overline{R}_{CS-ICDT}(c_j) &= \sum_{k=1}^{K} m_{jk}\overline{P}^{\mathcal{L}}_{IDM}(c_k), \quad \forall j = 1, 2, \dots, K. \end{aligned} \tag{5.23}$$

CS-ICDT applies the stochastic dominance criterion on these risk intervals for obtaining the non-dominated states set at that leaf node. According to that criterion, a class value c_j dominates another one c_k if, and only if, $\overline{R}_{CS-ICDT}(c_j) < \underline{R}_{CS-ICDT}(c_k), \quad \forall j, k = 1, 2, \dots, K$. Therefore, the non-dominated states set predicted by CS-ICDT at that terminal node is determined as follows:

$$nds^{\mathcal{L}}_{CS-ICDT} = \left\{c_j \mid \underline{R}_{CS-ICDT}(c_j) \leqslant \overline{R}_{CS-ICDT}(c_k), \quad \forall k = 1, 2, \dots, K\right\}, \tag{5.24}$$

where $\underline{R}_{CS-ICDT}(c_j)$ and $\overline{R}_{CS-ICDT}(c_j)$ are computed through Equation (5.23).

The procedure to classify an instance with CS-ICDT is summarized in Algorithm 8.

5.6.2 Naive Credal Classifier versus Imprecise Credal Decision Tree

An experimental analysis carried out by Abellán and Masegosa [10] showed that ICDT performs better than NCC since it achieves a better trade-off between informative and accurate predictions; even though ICDT makes more incorrect predictions than NCC, the former algorithm is much more informative than the latter.

Algorithm 8: Procedure to classify an instance with CS-ICDT.

```
Procedure Classify_CS-ICDT(ICDT $\mathcal{T}$, instance with attribute vector $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}\right)$).
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the attribute values $x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}$.
2. Compute the IDM probability intervals at $\mathcal{L}$, $\left\{\left[\underline{P}^{\mathcal{L}}_{IDM}(c_j), \overline{P}^{\mathcal{L}}_{IDM}(c_j)\right], \quad j = 1, 2, \dots, K\right\}$, through Equation (5.22).
3. Obtain the risk intervals from these probability intervals by means of Equation (5.23), $\left\{\left[\underline{R}_{CS-ICDT}(c_j), \overline{R}_{CS-ICDT}(c_j)\right], \quad j = 1, 2, \dots, K\right\}$.
4. Use the stochastic dominance criterion on the risk intervals for obtaining the non-dominated states set at $\mathcal{L}$:
       $$h^{CS-ICDT}(\mathbf{x}) = nds^{\mathcal{L}}_{CS-ICDT},$$
   where $nds^{\mathcal{L}}_{CS-ICDT}$ is determined via Equation (5.24).
return $h^{CS-ICDT}(\mathbf{x})$
```

The same conclusions were derived for the adaptations of NCC and ICDT for cost-sensitive scenarios (CS-NCC and CS-ICDT). Indeed, for all the error costs matrices considered in [10], CS-ICDT achieves a better trade-off between low misclassification cost and informative predictions than CS-NCC [10].

6 MULTI-LABEL CLASSIFICATION

6.1 Introduction

Traditional classification assumes that each instance has a single value of a class variable. This task has been successfully employed in practical applications. Nevertheless, there are domains where traditional classification does not fit well. For example, in *text categorization* [152, 185], a text can cover multiple topics simultaneously, such as *sports*, *Olympic Games*, and *France*[1]; within *biology*, both gens and proteins can have more than one function simultaneously [25, 29]; several emotions can appear in a *music* fragment [201]. In these domains, where each instance may have multiple labels simultaneously, the Multi-Label Classification task (MLC) is more suitable to be used.

MLC aims to learn a model that, for an instance described via a set of attributes or features, predicts the set of labels to which that instance belongs. The learned MLC model can also predict the posterior probability that a given instance belongs to each label, which leads to a label ranking for such an instance.

It should be noted that traditional classification is a particular case of MLC in which each instance only belongs to a unique label. Consequently, MLC is a much more complex task to solve than traditional classification. Indeed, in MLC, the number of label sets exponentially grows as there are more labels. In order to handle this issue, it is important that multi-label classifiers exploit label correlations. For this reason, many works have been carried out during the last years for exploiting correlations between labels in MLC. Examples can be found in [116, 138, 233, 234]. Moreover, in MLC, very few instances have often associated a certain label. Hence, many MLC datasets may suffer from a class-imbalance problem [50, 175]. For this reason, for many MLC algorithms, it might be difficult to predict that certain instances belong to some labels.

Evaluating the performance of a traditional classification method is direct, as we have shown in Section 4.2.1. However, in MLC, the evaluation is far more complicated because each instance might be associated with multiple

[1] In this context, “sports" means that the text is related to sports, “Olympic Games" indicates that the text is associated with the Olympic Games and “France" means that the text covers news in France.

labels simultaneously and, thus, an MLC evaluation metric can focus on the predicted label sets or the performance of the algorithm in each one of the labels. The MLC evaluation metrics that focus on the former issue are known as *instance-based metrics* [96, 99, 185] and the MLC evaluation metrics that focus on the latter issue are called *label-based metrics* [203].

Many MLC algorithms have been developed so far. Essentially, they can be divided into two groups [230]. On the one hand, the *problem transformation methods* convert the MLC task into multiple traditional classification problems and then combine the outputs of such problems to provide an output for the MLC task. On the other hand, the *algorithm adaptation methods* directly adapt the existing traditional classification algorithms for MLC.

The remainder of this chapter is structured as follows: Section 6.2 describes the Multi-Label Classification paradigm. The main evaluation metrics for Multi-Label Classification are exposed in Section 6.3. Sections 6.4 and 6.5 detail, respectively, the problem transformation methods and algorithm adaptation methods for Multi-Label Classification considered in this thesis work.

6.2 Multi-Label Classification

The Multi-Label Classification task (MLC) aims to predict, for an instance described by a set of attributes or features, the set of labels associated with such an instance.

Formally, the MLC problem starts from the following issues:

- A set of d predictive attributes $\left\{X^1, X^2, \ldots, X^d\right\}$. Let $\mathrm{Dom}(X^i)$ denote the domain of the X^i attribute, $\quad \forall i = 1, 2, \ldots, d$.

- A label set $\mathcal{Y} = \{y_1, y_2, \ldots, y_{n_L}\}$, where $n_L > 1$.

Definition 6.2.1 *When an instance belongs to a label y_j, with $j \in \{1, 2, \ldots, n_L\}$, it is said that y_j is relevant for such an instance. Otherwise, it is said that y_j is irrelevant for that instance.*

MLC aims to learn a model $h : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \rightarrow 2^{\mathcal{Y}}$ that, for a given instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, returns the set of labels that are predicted to be associated with such an instance, namely $h\left(\mathbf{x}\right)$.

Similarly to traditional classification, in MLC, a training set $\mathcal{D}_{train} = \left\{(\mathbf{x_j}, \mathbf{Y_j}), \quad j = 1, 2, \ldots, N_{tr}\right\}$ is used for learning the model. For the jth training instance, $\mathbf{x_j} = (x^1_{r_{1j}}, x^2_{r_{2j}}, \ldots, x^d_{r_{dj}})$ denotes its attribute vector, with $x^i_{r_{ij}} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, and $\mathbf{Y_j} \subseteq \mathcal{Y}$ its label set, $\quad \forall j = 1, 2, \ldots, N_{tr}$.

There are four measures for characterizing the properties of $\mathcal{D}_{tr}$[2]:

- **Label Cardinality**: The average number of labels per instance:

$$\mathrm{L\_Card}\left(\mathcal{D}_{train}\right) = \frac{1}{N_{tr}} \sum_{j=1}^{N_{tr}} \left|\mathbf{Y}_j\right|. \tag{6.1}$$

- **Label Density**: It indicates the average proportion of labels per instance and is obtained by normalizing the label cardinality by the number of labels:

$$\mathrm{L\_Dens}\left(\mathcal{D}_{train}\right) = \frac{\mathrm{L\_Card}\left(\mathcal{D}_{train}\right)}{n_L}. \tag{6.2}$$

- **Label diversity**: The number of distinct labels sets that appear in the set:

$$\mathrm{L\_Div}\left(\mathcal{D}_{train}\right) = \left|\mathbf{Y} \subseteq \mathcal{Y} \mid \exists j \in \{1, 2, \ldots, N_{tr}\} \text{ s.t } \quad \mathbf{Y} = \mathbf{Y}_j\right|. \tag{6.3}$$

- **Proportion label diversity**: It is the number of different label sets divided by the number of instances:

$$\mathrm{PL\_Div}\left(\mathcal{D}_{train}\right) = \frac{\mathrm{L\_Div}\left(\mathcal{D}_{train}\right)}{N_{tr}}. \tag{6.4}$$

Alternatively, in many cases, the learned model is described by means of a real-valued function $f : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \times \mathcal{Y} \to \mathbb{R}$ which, for a given instance with attribute vector $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, and a label $y_j \in \mathcal{Y}$, returns the predicted posterior probability that y_j is relevant for that instance, namely $f(\mathbf{x}, y_j)$.

For an instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, the real-valued function f yields a ranking function $rank_{f(\mathbf{x})} : \mathcal{Y} \to \{1, 2, \ldots, n_L\}$. It represents the predicted ranking of labels for the instance and is implicitly determined satisfying $rank_{f(\mathbf{x})}(y_j) < rank_{f(\mathbf{x})}(y_k) \quad \forall y_j, y_k$ such that $f(\mathbf{x}, y_j) > f(\mathbf{x}, y_k)$.

A threshold function $thr : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \to \mathbb{R}$ can be used to obtain the predicted set of relevant labels for an instance given the real-valued function f. For an instance with attribute vector $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, such a label set is extracted from f and thr in the following way:

$$h(\mathbf{x}) = \left\{y_j \in \mathcal{Y} \mid f(\mathbf{x}, y_j) > thr(\mathbf{x})\right\}. \tag{6.5}$$

There are three main options to calibrate the threshold function [230]:

[2] Such measures are also useful to characterize the properties of any multi-label dataset.

- Sometimes, the threshold function is fixed to a constant value, which is normally equal to 0.5. When all unseen test instances are available, the constant value can be set in such a way that the difference between the label cardinalities in the training and test sets is minimized [178].

- The second option consists of inducing the threshold function from the training instances [119]. For example, in some cases, the function t is assumed to be a linear model [86, 231].

- Finally, some algorithms have their own mechanism for determining the predicted set of relevant labels for an instance from the label ranking predicted for that instance. An example can be found in [94].

6.3 Evaluation metrics in Multi-Label Classification

Similarly to standard classification, in order to evaluate the performance of a multi-label classifier described through a set-valued function $h : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \to 2^{\mathcal{Y}}$ and a real-valued function $f : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \times \mathcal{Y} \to \mathbb{R}$, a test set $\mathcal{D}_{test}$ tends to be employed.

The evaluation metrics proposed so far for MLC can be divided into two groups: *instance-based* metrics [96, 99, 185] and *label-based* metrics [203]. The metrics of the former group evaluate the performance of a classifier in each test instance and then compute the average value across all test instances, whereas label-based metrics evaluate the performance of a classifier in each label similarly to binary classification and then compute the average value across all labels (macro averaging) or all instance/label pairs (micro averaging). Furthermore, both instance-based and label-based metrics can be divided into *classification-based* measures, which focus on the predicted label sets (set-valued function h), and *ranking-based* metrics, which focus on the predicted label rankings (real-valued function f).

Let $N_{test} = |\mathcal{D}_{test}|$ be the number of test instances. Let $x^i_{r_{ij}}$ denote the value of the jth test instance for the ith attribute, $\forall i = 1, 2, \ldots, d$, $\mathbf{x}_j = \left(x^1_{r_{1j}}, x^2_{r_{2j}}, \ldots, x^d_{r_{dj}}\right)$ its attribute vector, and $\mathbf{Y}_j \subseteq \mathcal{Y}$ its label set, $\forall j = 1, 2, \ldots, N_{test}$. We show below the main evaluation metrics in each one of the groups described above.

6.3.1 Instance-based metrics

1. **Classification**

- **Subset Accuracy**: The proportion of instances for which the predicted label set coincides with the set of relevant labels:

$$\mathrm{Subset\_Accuracy}(h) = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} [[h(\mathbf{x_i}) = \mathbf{Y}_i]], \qquad (6.6)$$

  where $[[h(\mathbf{x_i}) = \mathbf{Y}_i]]$ is equal to 1 if $h(\mathbf{x_i}) = \mathbf{Y}_i$ and 0 otherwise, $\forall i = 1, 2, \ldots, N_{test}$. This metric can be regarded as the extension of the Accuracy metric to MLC and is normally very strict, especially when the number of labels is very large.

- **Hamming Loss**: It indicates the proportion of pairs of label-instance incorrectly classified:

$$\mathrm{Hamming\_Loss}(h) = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} |h(\mathbf{x_i}) \triangle \mathbf{Y}_i|, \qquad (6.7)$$

  $\triangle$ being the symmetric difference between two sets, i.e, the elements belonging to one set but not to the other one.

- **Accuracy**: It consists of the average Jaccard similarity coefficient between the predicted label sets and the sets of relevant labels:

$$\mathrm{Accuracy}(h) = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \frac{|h(\mathbf{x_i}) \cap \mathbf{Y}_i|}{|h(\mathbf{x_i}) \cup \mathbf{Y}_i|}. \qquad (6.8)$$

- **Precision**: It indicates, between the labels predicted as relevant for an instance, the average proportion of them that are actually relevant for such an instance:

$$\mathrm{Precision}(h) = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \frac{|h(\mathbf{x_i}) \cap \mathbf{Y}_i|}{|h(\mathbf{x_i})|}. \qquad (6.9)$$

- **Recall**: It measures, between the labels that are associated with an instance, the average proportion of them that are predicted as relevant for that instance:

$$\mathrm{Recall}(h) = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \frac{|h(\mathbf{x_i}) \cap \mathbf{Y}_i|}{|\mathbf{Y}_i|}. \qquad (6.10)$$

- **F1**: The harmonic mean between Precision and Recall:

$$\mathrm{F1}(h) = \frac{2 \times \mathrm{Precision}(h) \times \mathrm{Recall}(h)}{\mathrm{Precision}(h) + \mathrm{Recall}(h)}. \tag{6.11}$$

2. **Ranking**

- **One Error**: It measures the proportion of instances for which the label with the highest predicted posterior probability is irrelevant:

$$\mathrm{One\_Error}(f) = \frac{1}{N_{Test}} \sum_{i=1}^{N_{Test}} \arg \max_{j=1,2,\dots,n_L} f(\mathbf{x_i}, y_j) \notin \mathbf{Y}_i. \tag{6.12}$$

- **Coverage**: It consists of the average number of steps that are required to go down the label ranking for covering all relevant labels for an instance:

$$\mathrm{Coverage}(f) = \frac{1}{N_{Test}} \sum_{i=1}^{N_{Test}} \max_{y_j \in \mathbf{Y_i}} \mathrm{rank}_{f(\mathbf{x_i})}(y_j) - 1. \tag{6.13}$$

- **Ranking Loss**: It indicates the average proportion of pairs of relevant-irrelevant labels reversely ordered:

$$\mathrm{Ranking\_Loss}(f) = \frac{1}{N_{Test}} \sum_{i=1}^{N_{Test}} \frac{|\mathbf{Z}_i|}{|\mathbf{Y}_i| \times \left|\overline{\mathbf{Y}_i}\right|}, \tag{6.14}$$

where $\overline{\mathbf{Y}_i}$ is the complement of $\mathbf{Y}_i$ and
$\mathbf{Z}_i = \left\{(y_j, y_k) \mid \mathrm{rank}_{f(\mathbf{x_i})}(y_j) > \mathrm{rank}_{f(\mathbf{x_i})}(y_k), \quad y_j \in \mathbf{Y}_i, y_k \in \overline{\mathbf{Y}_i}\right\}$,
$\forall i = 1, 2, \dots, N_{test}$.

- **Average Precision**: The average proportion of labels with a higher predicted posterior probability than a relevant label:

  Formally, for each $i = 1, 2, \dots, N_{Test}$, and $y_j \in \mathbf{Y}_i$, let us consider $\Lambda_{i,j} = \left\{y_k \mid \mathrm{rank}_{f(\mathbf{x_i})}(y_k) \leqslant \mathrm{rank}_{f(\mathbf{x_i})}(y_j),\ 1 \leqslant k \leqslant n_L\right\}$, $\mathrm{rank}_{f(\mathbf{x_i})}$ being the ranking function derived from f for the ith test instance. Average Precision is defined as follows:

$$\mathrm{Average\_Precision} = \frac{1}{N_{Test}} \sum_{i=1}^{N_{Test}} \frac{1}{|\mathbf{Y}_i|} \sum_{y_j \in \mathbf{Y}_i} \frac{|\Lambda_{i,j}|}{\mathrm{rank}_{f(\mathbf{x}_i)}(y_j)}. \tag{6.15}$$

6.3.2 Label-based metrics

1. **Classification**

For each label $y_j \in \mathcal{Y}$, the number of true positives (TP_j), true negatives (TN_j), false positives, (FP_j), and false negatives (FN_j), are considered for the evaluation measures based on label classification:

$$\begin{aligned} TP_j(h) &= \left|\left\{i \in \{1,2,\ldots,N_{test}\} : y_j \in \mathbf{Y}_i \wedge y_j \in h(\mathbf{x_i})\right\}\right|, \\ TN_j(h) &= \left|\left\{i \in \{1,2,\ldots,N_{test}\} : y_j \notin \mathbf{Y}_i \wedge y_j \notin h(\mathbf{x_i})\right\}\right|, \\ FP_j(h) &= \left|\left\{i \in \{1,2,\ldots,N_{test}\} : y_j \notin \mathbf{Y}_i \wedge y_j \in h(\mathbf{x_i})\right\}\right|, \\ FN_j(h) &= \left|\left\{i \in \{1,2,\ldots,N_{test}\} : y_j \in \mathbf{Y}_i \wedge y_j \notin h(\mathbf{x_i})\right\}\right|. \end{aligned} \tag{6.16}$$

The label classification-based evaluation measures proposed so far are based on the micro/macro averaging of the evaluation metrics for binary classification based on the previous indicators. As explained before, micro averaging consists of averaging across all instance-label pairs, while macro averaging consists of averaging overall labels.

Let Acc_j, $Prec_j$, Rec_j and $F1_j$ denote, respectively, the Accuracy, Precision, Recall, and F1 for the label y_j:

$$\begin{aligned} Acc_j(h) &= \frac{TP_j(h) + TN_j(h)}{TP_j(h) + TN_j(h) + FP_j(h) + FN_j(h)}, \\ Prec_j(h) &= \frac{TP_j(h)}{TP_j(h) + FP_j(h)}, \quad Rec_j(h) = \frac{TP_j(h)}{TP_j(h) + FN_j(h)}, \\ F1_j(h) &= \frac{2 \times Prec_j(h) \times Rec_j(h)}{Prec_j(h) + Rec_j(h)}, \quad \forall j = 1,2,\ldots,n_L. \end{aligned} \tag{6.17}$$

We show below the main label classification-based evaluation metrics:

- **Micro Accuracy**: It corresponds to the Accuracy averaged overall instance-label pairs:

$$Micro\_Accuracy(h) = \frac{\sum_{j=1}^{n_L} \left(TP_j(h) + TN_j(h)\right)}{\sum_{j=1}^{n_L} \left(TP_j(h) + TN_j(h) + FP_j(h) + FN_j(h)\right)}. \tag{6.18}$$

- **Macro Accuracy**: The Accuracy averaged across all labels:

$$Macro\_Accuracy(h) = \frac{1}{n_L} \sum_{j=1}^{n_L} Acc_j(h). \tag{6.19}$$

- **Micro Precision**: It consists of the average Precision overall instance-label pairs:

$$\text{Micro\_Precision}(h) = \frac{\sum_{j=1}^{n_L} TP_j(h)}{\sum_{j=1}^{n_L} \left(TP_j(h) + FP_j(h)\right)}. \tag{6.20}$$

- **Macro Precision**: The average Precision across all labels:

$$\text{Macro\_Precision}(h) = \frac{1}{n_L} \sum_{j=1}^{n_L} \text{Prec}_j(h). \tag{6.21}$$

- **Micro Recall**: It indicates the Recall averaged across all instance-label pairs:

$$\text{Micro\_Recall}(h) = \frac{\sum_{j=1}^{n_L} TP_j(h)}{\left(\sum_{j=1}^{n_L} TP_j(h) + FN_j(h)\right)}. \tag{6.22}$$

- **Macro Recall**: The average Recall across all labels:

$$\text{Macro\_Recall}(h) = \frac{1}{n_L} \sum_{j=1}^{n_L} \text{Rec}_j(h). \tag{6.23}$$

- **Micro F1**: It is the harmonic mean between Micro Precision and Micro Recall:

$$\text{Micro\_F1}(h) = \frac{2 \times \text{Micro\_Precision}(h) \times \text{Micro\_Recall}(h)}{\text{Micro\_Precision}(h) + \text{Micro\_Recall}(h)}. \tag{6.24}$$

- **Macro F1**: The F1 averaged across all labels:

$$\text{Macro\_F1}(h) = \frac{1}{n_L} \sum_{j=1}^{n_L} \text{F1}_j(h). \tag{6.25}$$

2. **Ranking**

- **Micro AUC**: It corresponds to the AUC measure across all instance-label pairs. Let $\mathcal{S}^+$ and $\mathcal{S}^-$ denote, respectively, the sets of pairs of instance-relevant label and instance-irrelevant label in the test set:

$$\begin{aligned} \mathcal{S}^+ &= \{(\mathbf{x_i}, y) \mid y \in \mathbf{Y_i}, \quad i \in \{1, 2, \dots, N_{test}\}\}, \\ \mathcal{S}^- &= \{(\mathbf{x_i}, y) \mid y \notin \mathbf{Y_i}, \quad i \in \{1, 2, \dots, N_{test}\}\}, \end{aligned} \tag{6.26}$$

Micro AUC is defined in the following way:

$$\mathrm{Micro\_AUC}(f) = \frac{|\mathcal{Z}(f)|}{|\mathcal{S}^+|\,|\mathcal{S}^-|}, \tag{6.27}$$

where

$$\mathcal{Z}(f) = \big\{(\mathbf{x}',\mathbf{x}'',y',y'') \mid f(\mathbf{x}',y') > f(\mathbf{x}'',y''), \\ (\mathbf{x}',y') \in \mathcal{S}^+ \wedge (\mathbf{x}'',y'') \in \mathcal{S}^-)\big\}.$$

- **Macro AUC**: The average AUC overall labels. Formally, for each $y_j \in \mathcal{Y}$, let $\mathcal{Z}_j$ ($\overline{\mathcal{Z}_j}$) be the set of test instances for which y_j is relevant (irrelevant):

$$\begin{aligned} \mathcal{Z}_j &= \big\{i \in \{1,2,\ldots,N_{test}\} \mid y_j \in \mathbf{Y_i}\big\}, \\ \overline{\mathcal{Z}_j} &= \big\{i \in \{1,2,\ldots,N_{test}\} \mid y_j \notin \mathbf{Y_i}\big\}. \end{aligned} \tag{6.28}$$

For each label $y_j \in \mathcal{Y}$, the AUC is determined by:

$$\mathrm{AUC}_j(f) = \frac{\big|\big\{(i,k) : f(\mathbf{x_i},y_j) \geqslant f(\mathbf{x_k},y_j), \quad i \in \mathcal{Z}_j \wedge k \in \overline{\mathcal{Z}_j}\big\}\big|}{\big|\mathcal{Z}_j\big|\,\big|\overline{\mathcal{Z}_j}\big|}. \tag{6.29}$$

Then, Macro AUC is computed as follows:

$$\mathrm{Macro\_AUC}(f) = \frac{1}{n_L}\sum_{j=1}^{n_L} \mathrm{AUC}_j(f). \tag{6.30}$$

6.4 Problem transformation methods

As said previously, the problem transformation methods convert the MLC task into multiple traditional classification problems and combine their outcomes to provide an outcome for the MLC problem. In this work thesis, we consider three algorithms belonging to this category. Two of them consider a binary classification task per label, and the other one a binary classification problem for each pair of labels. We detail all these methods below.

6.4.1 Binary Relevance

The Binary Relevance method (BR) [36] is probably the simplest approach to MLC. It decomposes this task into multiple independent binary classification problems, one per label. For classifying a new instance, its predicted set of relevant labels, as well as the predicted posterior probabilities of the relevance of the labels for that instance, are directly obtained from such learned classifiers.

Formally, for each label $y_j \in \mathcal{Y}$, BR learns a binary classifier $h_j^{BR} : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots$ $\{0, 1\}$. Such a classifier uses the same predictive attributes as the original MLC problem. The class variable indicates whether y_j is relevant or irrelevant for an instance. In order to learn the classifier h_j^{BR}, the following training set is considered:

$$\mathcal{D}_j^{BR} = \left\{ (\mathbf{x_i}, \phi\left(\mathbf{Y_i}, y_j\right)), \quad i = 1, 2, \ldots, N_{tr} \right\}, \tag{6.31}$$

where $\phi\left(\mathbf{Y_i}, y_j\right)$ indicates the relevance of y_j for the ith training instance, i.e,

$$\phi\left(\mathbf{Y_i}, y_j\right) = \left\{ \begin{array}{lll} 1 & \text{if} & y_j \in \mathbf{Y_i} \\ & & \\ 0 & \text{if} & y_j \notin \mathbf{Y_i} \end{array} \right\}, \tag{6.32}$$

$\forall i = 1, 2, \ldots, N_{tr}, \quad j = 1, 2, \ldots, n_L.$

The classifier h_j^{BR} is learned by employing a binary classification algorithm $\mathcal{B}$ on $\mathcal{D}_j^{BR}$ ($h_j^{BR} \leftarrow \mathcal{B}(\mathcal{D}_j^{BR})$). The algorithm $\mathcal{B}$ is known as the *base classifier* of BR. In addition, the binary classifier can also return a real-valued function $f_j^{BR} : \left(\mathrm{Dom}(X^1), \mathrm{Dom}(X^2), \ldots, \mathrm{Dom}(X^d)\right) \to \mathbb{R}$, which, for a given instance, returns the predicted posterior probability that y_j is relevant for that instance.

In order to classify an instance with attribute vector $\mathbf{x} = \left(x_{r_1}^1, x_{r_2}^2, \ldots, x_{r_d}^d\right)$, where $x_{r_i}^i \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, the predicted set of relevant labels for that instance is composed of those labels predicted as relevant by the corresponding binary classifier:

$$h^{BR}(\mathbf{x}) = \left\{ y_j \mid h_j^{BR}(\mathbf{x}) = 1, \quad j \in \{1, 2, \ldots, n_L\} \right\}. \tag{6.33}$$

The predicted posterior probabilities about the relevance of the labels for the instance are also directly obtained from the posterior probabilities predicted by the binary classifiers:

$$f^{BR}(\mathbf{x}, y_j) = f_j^{BR}(\mathbf{x}), \quad \forall j = 1, 2, \ldots, n_L. \tag{6.34}$$

We must remark the following issues about BR:

- Despite being very simple, BR has achieved good results in practice, comparable with more sophisticated MLC algorithms [144].
- However, BR has an important drawback: as it assumes that all labels are independent, BR ignores correlations among the labels, which are quite common.

- Furthermore, the binary classifiers of BR tend to suffer from a class-imbalance problem because, as explained before, very few instances have often associated a label in MLC.

6.4.2 Classifier Chains

The Classifier Chain algorithm (CC) [178] also considers a binary classifier per label. Nonetheless, unlike BR, in CC, the previous labels according to an established order are used as additional predictive attributes. Also, in order to classify an instance, for each classifier, the predictions made by the predecessor classifiers according to the established order are taken into account.

Formally, let $\sigma : \{1, \ldots, n_L\} \rightarrow \{1, \ldots, n_L\}$ be a permutation that leads to a label order $y_{\sigma(1)} \succ y_{\sigma(2)} \succ \ldots \succ y_{\sigma(n_L)}$. For the jth label, $y_{\sigma(j)}$, a binary classifier
$h^{CC}_{\sigma(j)} : \Big(\left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right), \{0,1\} \times \overset{j-1}{\cdots} \times \{0,1\}\Big) \rightarrow \{0,1\}$ is learned. Such a classifier uses, as the predictive attributes, the original attribute space and the predecessor labels to $y_{\sigma(j)}$ according to σ. The class variable indicates the relevance of $y_{\sigma(j)}$ for an instance. In order to learn the classifier $h^{CC}_{\sigma(j)}$, the following training set is considered:

$$\mathcal{D}^{CC}_{\sigma(j)} = \left\{ \left([\mathbf{x_i}, pred_i(\sigma(j))], \phi\left(\mathbf{Y_i}, y_{\sigma(j)}\right) \right), \quad i = 1, 2, \ldots, N_{tr} \right\}, \tag{6.35}$$

where, for each $i = 1, 2, \ldots, N_{tr}, \quad j = 1, 2, \ldots, n_L$,
$pred_i(\sigma(j)) = \left(\phi\left(\mathbf{Y_i}, y_{\sigma(1)}\right), \phi\left(\mathbf{Y_i}, y_{\sigma(2)}\right), \ldots, \phi\left(\mathbf{Y_i}, y_{\sigma(j-1)}\right)\right)$ and
$\phi\left(\mathbf{Y_i}, y_{\sigma(j)}\right)$ is determined via Equation (6.32).

In this way, $h^{CC}_{\sigma(j)}$ utilizes the original training instances considering, for each one of them, its attribute values and the relevance of the predecessor labels to $y_{\sigma(j)}$ for it. The class value corresponds to the relevance of $y_{\sigma(j)}, \quad \forall j = 1, 2, \ldots, n_L$.

Similarly to BR, a binary classification algorithm $\mathcal{B}$ is employed on $\mathcal{D}^{CC}_{\sigma(j)}$ to learn the classifier $h^{CC}_{\sigma(j)}$ ($h^{CC}_{\sigma(j)} \leftarrow \mathcal{B}(\mathcal{D}^{CC}_{\sigma(j)})$). The algorithm $\mathcal{B}$ is also called the *base classifier* of CC. As in BR, the classifier can also return a real-valued function
$f^{CC}_{\sigma(j)} : \Big(\left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right), \{0,1\} \times \overset{j-1}{\cdots} \times \{0,1\}\Big) \rightarrow \{0,1\}$ that, for a given instance, outputs the predicted posterior probability about the relevance of $y_{\sigma(j)}$ for that instance, $\quad \forall j = 1, 2, \ldots, n_L$.

Given an instance to classify whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in Dom(X^i) \quad \forall i = 1, 2, \ldots, d$, the predictions on the binary classifiers are made by following the order established by the permutation σ. Each classifier takes into account the predictions made by the predecessor classifiers according to σ:

$$\begin{aligned}
&\lambda_{y_{\sigma(1)}}(\mathbf{x}) = h^{CC}_{\sigma(1)}(\mathbf{x}), \\
&\lambda_{y_{\sigma(j)}}(\mathbf{x}) = h^{CC}_{\sigma(j)}\left(\mathbf{x}, \lambda_{y_{\sigma(1)}}(\mathbf{x}), \dots, \lambda_{y_{\sigma(j-1)}}(\mathbf{x})\right), \quad \forall j = 2, \dots, n_L.
\end{aligned} \tag{6.36}$$

The set of labels predicted as relevant for the instance is composed of those labels predicted as relevant by the corresponding binary classifier:

$$h^{CC}(\mathbf{x}) = \left\{ y_{\sigma(j)} \mid \lambda_{y_{\sigma(j)}}(\mathbf{x}) = 1, \quad 1 \leqslant j \leqslant n_L \right\}. \tag{6.37}$$

The predicted posterior probabilities about the relevance of the labels for the instance are derived in a straightforward way from the posterior probabilities predicted by the binary classifiers:

$$\begin{aligned}
&f^{CC}_{\sigma(1)}(\mathbf{x}, y_{\sigma(1)}) = f^{CC}_{\sigma(1)}(\mathbf{x}), \\
&f^{CC}_{\sigma(j)}(\mathbf{x}, y_{\sigma(j)}) = f^{CC}_{\sigma(j)}\left(\mathbf{x}, \lambda_{y_{\sigma(1)}}(\mathbf{x}), \dots, \lambda_{y_{\sigma(j-1)}}(\mathbf{x})\right), \quad \forall j = 2, \dots, n_L.
\end{aligned} \tag{6.38}$$

We must remark the following points about CC:

- Unlike BR, CC exploits correlations among labels as, for making predictions about a label, it considers other labels. Indeed, CC is considered a simple and effective method to exploit label correlations in MLC.

- Moreover, CC is one of the methods that achieved the best results in a comparative experimental study about MLC algorithms carried out in [144].

- Nevertheless, the label order strongly influences the performance of CC, and there is no way to determine the optimal label order so far.

- As in BR, the binary classifiers of CC also tend to suffer from a class-imbalance problem.

6.4.2.1 *Label order in Classifier Chain*

Several approaches have been developed during the last years for determining a suitable label order in CC. Since the number of possible label orders enormously increases as the number of labels is higher (the factorial of n_L), it is not viable to exhaustively explore the complete search space of the possible label orders.

- Genetic algorithms for label ordering were developed in [100, 101]. We must remark that such algorithms use a wrapper approach to evaluate the goodness of each candidate order, which implies that they train a CC for evaluating each candidate. Thus, the computational cost of these methods is enormous.

- Most of the label ordering procedures developed during the last years previously estimate correlations between labels. For instance, in [124], several greedy procedures to insert the labels in the chain were proposed. They are based on the entropies of the candidate labels conditioned on the ones not inserted yet. Specifically, for each pair of labels $\{y_j, y_k\} \subseteq \mathcal{Y}$, they consider the entropy of y_j conditioned on y_k:

$$S(y_j \mid y_k) = \frac{n_{tr}(y_k)}{N_{tr}} S(y_j \mid y_k = 1) + \frac{n_{tr}(\overline{y_k})}{N_{tr}} S(y_j \mid y_k = 0), \qquad (6.39)$$

where $n_{tr}(y_k)$ and $n_{tr}(\overline{y_k})$ are the number of training instances for which y_k is relevant and irrelevant, respectively, and $S(y_j \mid y_k = 1)$ ($S(y_j \mid y_k = 0)$) is the entropy of y_j on the subset of the training set composed of those instances for which y_j is relevant (irrelevant):

$$\begin{aligned} S(y_j \mid y_k = 1) = &-\frac{n_{tr}(y_j, y_k)}{n_{tr}(y_k)} \log_2\left(\frac{n_{tr}(y_j, y_k)}{n_{tr}(y_k)}\right) - \\ &\frac{n_{tr}(\overline{y_j}, y_k)}{n_{tr}(y_k)} \log_2\left(\frac{n_{tr}(\overline{y_j}, y_k)}{n_{tr}(y_k)}\right), \\ S(y_j \mid y_k = 0) = &-\frac{n_{tr}(y_j, \overline{y_k})}{n_{tr}(\overline{y_k})} \log_2\left(\frac{n_{tr}(y_j, \overline{y_k})}{n_{tr}(\overline{y_k})}\right) - \\ &\frac{n_{tr}(\overline{y_j}, \overline{y_k})}{n_{tr}(\overline{y_k})} \log_2\left(\frac{n_{tr}(\overline{y_j}, \overline{y_k})}{n_{tr}(\overline{y_k})}\right), \end{aligned} \qquad (6.40)$$

$n_{tr}(y_j, y_k)$ being the number of training instances that have associated both y_j and y_k, $n_{tr}(\overline{y_j}, y_k)$ the number of training instances that have associated y_k but not y_j, $n_{tr}(y_j, \overline{y_k})$ the number of training instances for which y_j is relevant but y_k irrelevant, and $n_{tr}(\overline{y_j}, \overline{y_k})$ the number of training instances for which both y_j and y_k are irrelevant, $\quad \forall j, k = 1, 2, \ldots, n_L$.

The greedy procedures proposed in [124] are based on the following idea: If $S(y_j \mid y_k) \leqslant S(y_k \mid y_j)$, then y_j should be placed before y_k in the chain, $\quad \forall j, k = 1, 2, \ldots, n_L$. There are four greedy procedures based on this issue. We summarize below how the labels are selected at each step in each one of such procedures:

1. Compute, for each candidate label to insert, the sum of the entropies of the labels not inserted yet conditioned on the candidate label. The label with the lowest sum is placed after the remaining candidates.
2. For each candidate label to insert, compute the sum of the entropies of the labels not inserted yet conditioned on the candidate label. The label with the highest sum is placed before the remaining candidate labels.
3. Compute, for each candidate label, the sum of the entropies of that label conditioned on the ones not inserted yet. The label with the lowest sum is placed before the remaining candidates.
4. For each candidate label to insert, compute the sum of the entropies of such a label conditioned on the ones not inserted yet. The label with the highest sum is placed after the remaining candidates.

An experimental analysis carried out in [124] showed that CC with the aforementioned greedy procedures outperform other generalizations of CC that utilize more sophisticated methods to model label dependencies. Among these methods, we can mention Probabilistic Classifier Chain [73], which randomly determines the label order and, for classifying an instance, the label combination with highest joint probability distribution is selected; or the algorithms proposed in [87, 134], where the chain of classifiers is replaced by a Directed Acyclic Graph. Moreover, in [124], it was shown that the procedure 1 generally achieves better results than the other ones, although the differences are not statistically significant. It was not found a reason for this point.

- Very recently, in [214], a new label ordering algorithm has been proposed. It considers, for each pair of labels, a score about the correlation among them based on the ReliefF method [126]. The idea of ReliefF is that two labels are more correlated as they are more useful to separate the instances for which other labels are relevant or irrelevant. Afterwards, a threshold is used to select, for each label, the set of labels correlated with it. A greedy procedure is carried out to insert the labels in the chain. At each step of such a procedure, among the labels correlated with at least one label already inserted, the label with the highest number of candidate labels correlated with it is chosen. If there is no label correlated with the labels already inserted, the label with the highest number of candidate labels correlated with it is chosen. The selected label is placed before the remaining candidates in the chain.

A drawback of this method is that it is difficult to determine the threshold for obtaining the set of labels correlated with each label. Moreover, the developers of this method showed that, even though it improves the original CC algorithm, their proposal is not very effective unless, for each label, the features not correlated with it are removed. Hence, they proposed a feature selection procedure, also based on the ReliefF method, to eliminate, for each label, the labels and features not correlated with it. Nevertheless, we must remark that such a feature selection algorithm applied for each label implies a quite high computational time.

- Several methods that consider different label orders in CC have been proposed in the literature. An example is the Ensemble of Classifier Chains algorithm (EnsembleCC) [178], which we detail in Section 6.4.2.2. In [228], the Bayesian Classifier Chains method (Bayesian CC) was proposed. Such an algorithm uses a tree structure to model label dependencies. For each label, it considers an order in which that label is employed as the node of the dependencies tree. A CC is trained for each label order. To classify an instance, Bayesian CC combines the predictions made by the trained CCs by means of a majority vote. We must remark that, as EnsembleCC and Bayesian CC train multiple CCs, the computational times of these methods might be very high.

6.4.2.2 *Ensemble of Classifier Chains*

As the performance of the CC method is strongly influenced by the label order, the same developers of CC proposed the Ensemble of Classifier Chains algorithm (EnsembleCC) [178] to handle this issue.

EnsembleCC considers n_orders permutations $\sigma_1, \sigma_2, \ldots, \sigma_{n\_orders}$, where $\sigma_i : \{1, 2, \ldots, n_L\} \to \{1, 2, \ldots, n_L\} \quad \forall i = 1, 2, \ldots, n\_orders$. For each permutation σ_i, it considers a sample of the training set, $\mathcal{D}^i_{tr}$, with replacement ($\left|\mathcal{D}^i_{tr}\right| = |\mathcal{D}_{tr}|$), or without replacement ($\left|\mathcal{D}^i_{tr}\right| = \frac{2}{3}\,|\mathcal{D}_{tr}|$). A multi-label classifier $h^i : \left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right) \to 2^{\mathcal{Y}}$ is learned using $\mathcal{D}^i_{tr}$ as the training set and the order established by the permutation σ_i via the building procedure of CC. Such a classifier is often described via a real-valued function $f^i : \left(\left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right), \times \mathcal{Y}\right) \to \mathbb{R}$ that, given an instance and a label, returns the predicted posterior probability that such a label is relevant for that instance.

For classifying an instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in Dom(X_i) \quad \forall i = 1, 2, \ldots, d$, the predicted posterior probability that a label is relevant for that instance is computed through the average of the posterior probabilities about the relevance of such a label for the instance predicted by the CCs of the ensemble:

$$f^{ECC}(\mathbf{x}, y_j) = \frac{1}{n\_orders} \sum_{i=1}^{n\_orders} f^i(\mathbf{x}, y_j), \quad \forall j = 1, 2, \ldots, n_L. \tag{6.41}$$

The predicted set of relevant labels for the instance, namely $h^{ECC}(\mathbf{x})$, can be determined in two ways:

- Firstly, a label can be predicted as relevant if, and only if, it is predicted as relevant by the majority of the CCs:

$$\begin{aligned} h^{ECC}(\mathbf{x}) = \Bigg\{ y_j \mid & \sum_{i=1}^{n\_orders} [[y_j \in h^i(\mathbf{x})]] > \\ & \sum_{i=1}^{n\_orders} [[y_j \notin h^i(\mathbf{x})]], \quad j \in \{1, 2, \ldots, n_L\} \Bigg\}. \end{aligned} \tag{6.42}$$

- Secondly, a label can be predicted as relevant for the instance if, and only if, the average of the posterior probabilities predicted by the CCs of the ensemble is higher than 0.5:

$$h^{ECC}(\mathbf{x}) = \left\{ y_j \mid f^{ECC}(\mathbf{x}, y_j) > 0.5, \quad j \in \{1, 2, \ldots, n_L\} \right\}. \tag{6.43}$$

6.4.3 Calibrated Label Ranking

The Calibrated Label Ranking method (CLR) [94] transforms the MLC problem into a label ranking task. In order to determine the label ranking for an instance, CLR computes a score for each label via pairwise comparisons.

Specifically, for each pair of labels $\{y_j, y_k\} \subseteq \mathcal{Y}$, CLR learns a binary classifier $h_{jk}^{CLR} : \left(Dom(X^1), Dom(X^2), \ldots, Dom(X^d)\right) \to \{0, 1\}$. Such a classifier employs the same predictive attributes as the original MLC problem. The class variable indicates which of y_j and y_k is more relevant for an instance, that is, the relative relevance of y_j versus y_k for such an instance. For learning the classifier h_{jk}^{CLR}, the instances of the original training set that have associated one of the two labels but not the other one are considered, taking, for each one of them, its attribute values and which of the two labels is relevant for it. Hence, the following training set is considered for h_{jk}^{CLR}:

$$\mathcal{D}_{jk}^{CLR} = \left\{ \left(\mathbf{x_i}, \phi\left(\mathbf{Y_i}, y_j\right)\right) \mid \phi\left(\mathbf{Y_i}, y_j\right) \neq \phi\left(\mathbf{Y_i}, y_k\right), \quad 1 \leqslant i \leqslant N_{tr} \right\}. \tag{6.44}$$

Similarly to the other problem transformation methods considered in this work thesis, a binary classification algorithm $\mathcal{B}$ is utilized on $\mathcal{D}_{jk}^{CLR}$ to learn h_{jk}^{CLR} ($h_{jk}^{CLR} \leftarrow \mathcal{B}(\mathcal{D}_{jk}^{CLR})$). The algorithm $\mathcal{B}$ is called the *base classifier* of CLR.

When it is required to classify an instance with attribute vector $\mathbf{x} = (x_{r_1}^1, x_{r_2}^2, \ldots, x_{r_d}^d)$, where $x_{r_i}^i \in Dom(X^i) \quad \forall i = 1, 2, \ldots, d$, the relative relevances are predicted on the learned classifiers. For each label, the number of votes in the classifiers that involve such a label is considered:

$$num\_vot_{\mathbf{x}}(y_j) = \sum_{k=1}^{j-1} \left[\left[h_{kj}^{CLR}(\mathbf{x}) = 0\right]\right] + \sum_{k=j+1}^{n_L} \left[\left[h_{jk}^{CLR}(\mathbf{x})) = 1\right]\right], \quad \forall j = 1, 2, \ldots, n_L. \tag{6.45}$$

This leads to a label ranking for the instance. For distinguishing between relevant and irrelevant labels, CLR introduces a virtual label y_0, in such a way that the labels ranked above (below) y_0 are predicted as relevant (irrelevant). For each label y_j, a binary classifier $h_{j0}^{CLR} : (Dom(X^1), Dom(X^2), \ldots, Dom(X^d)) \to \{0, 1\}$ is learned, which predicts whether an instance has associated y_j or, equivalently, the relative relevance of y_j versus y_0 for such an instance. In order to learn h_{j0}^{CLR}, all training instances are considered, taking, for each one of them, its attribute values and the relevance of y_j for it. Formally, the following training set is used for h_{j0}:

$$\mathcal{D}_{j0}^{CLR} = \left\{ \left(\mathbf{x_i}, \phi\left(\mathbf{Y}_i, y_j\right)\right), \quad i = 1, 2, \ldots, N_{tr} \right\}. \tag{6.46}$$

The binary classification algorithm $\mathcal{B}$ employed for the classifiers corresponding to the pairwise comparisons is used to learn h_{j0}^{CLR} from $\mathcal{D}_{j0}^{CLR}$ ($h_{j0}^{CLR} \leftarrow \mathcal{B}(\mathcal{D}_{j0}^{CLR})$).

CLR also considers the number of votes of the virtual label for the instance to classify:

$$votes\_virtual\,(\mathbf{x}) = \sum_{j=1}^{n_L} \left[\left[h_{j0}^{CLR}(\mathbf{x}) = 0\right]\right]. \tag{6.47}$$

In addition, for each label, the number of votes is incremented in one if that label is predicted to be more relevant than the virtual one for the instance:

$$final\_votes_{\mathbf{x}}(y_j) = num\_vot_{\mathbf{x}}(y_j) + \left[\left[h_{j0}^{CLR}(\mathbf{x}) = 1\right]\right], \quad \forall j = 1, 2, \ldots, n_L. \tag{6.48}$$

The posterior probability predicted by CLR about the relevance of the label y_j for the instance is obtained by dividing the final number of votes, determined through Equation (6.48), by the number of labels:

$$f^{CLR}(\mathbf{x}, y_j) = \frac{final\_votes_{\mathbf{x}}(y_j)}{n_L}, \quad \forall j = 1, 2, \ldots, n_L. \tag{6.49}$$

Finally, the set of labels predicted by CLR as relevant for the instance is composed of those labels for which the final number of votes is higher than the number of votes of the virtual label:

$$h^{CLR}(\mathbf{x}) = \left\{ y_j \mid final\_votes_{\mathbf{x}}(y_j) > votes\_virtual\,(\mathbf{x})\,, \quad 1 \leqslant j \leqslant n_L \right\}. \tag{6.50}$$

We must remark the following points about CLR:

- Since CLR considers a binary classifier for each pair of labels, it allows exploiting pairwise label correlations.
- Furthermore, for the binary classifiers of CLR associated with the pairwise comparisons, only the training instances for which one of the two labels is relevant and the other one irrelevant are considered. Therefore, CLR alleviates the class-imbalance problem that frequently appears in MLC.
- Nonetheless, the number of classifiers learned by CLR quadratically grows as there are more labels. In contrast, with BR and CC, the growing of the number of learned classifiers with the number of labels is linear. Consequently, the main drawback of the CLR method is the computational time.

6.5 Algorithm adaptation methods

Several traditional classification algorithms were adapted for MLC. In this work thesis, we focus on the adaptations of Decision Trees [56] (Section 6.6) and the Nearest Neighbors algorithm [232] (Section 6.7).

6.6 Multi-Label Decision Tree

Decision Trees were adapted for MLC by Clare and King [56]. Such an adaptation is known as Multi-Label Decision Tree (ML-DT). For the split criterion, in each node, ML-DT needs to represent the uncertainty-based information about the label set, unlike Decision Trees for standard classification, which represent the uncertainty-based information about the class variable in that node. Furthermore, at leaf nodes, ML-DT must predict a set of labels, whereas Decision Trees for traditional classification predict a single value of the class variable.

Let $\mathcal{D}$ be the subset of the training set associated with a certain node. Let $N^{\mathcal{D}}$ denote the number of instances in $\mathcal{D}$, i.e, $N^{\mathcal{D}} = |\mathcal{D}|$. For each label $y_j \in \mathcal{Y}$, the Shannon entropy of y_j on $\mathcal{D}$ is considered:

$$S^{\mathcal{D}}(y_j) = -\frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}} \log_2\left(\frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}}\right) - \frac{n^{\mathcal{D}}(\overline{y_j})}{N^{\mathcal{D}}} \log_2\left(\frac{n^{\mathcal{D}}(\overline{y_j})}{N^{\mathcal{D}}}\right), \qquad (6.51)$$

$n^{\mathcal{D}}(y_j)$ being the number of instances in $\mathcal{D}$ that have associated y_j and $n^{\mathcal{D}}(\overline{y_j})$ the number of instances in $\mathcal{D}$ for which y_j is irrelevant.

The split criterion of ML-DT is based on the entropy of the label set $\mathcal{Y}$, which is computed by means of the sum of the entropies of the labels:

$$S^{\mathcal{D}}(\mathcal{Y}) = \sum_{j=1}^{n_L} S^{\mathcal{D}}(y_j). \qquad (6.52)$$

Similarly to most of the Decision Trees for standard classification, ML-DT employs a split criterion that consists of the gain of information about the label set given an attribute. Such a split criterion, for an attribute X^i whose possible values are $\{x_1^i, \ldots, x_{t_i}^i\}$, is defined as follows:

$$IG^{\mathcal{D}}(\mathcal{Y}, X^i) = S^{\mathcal{D}}(\mathcal{Y}) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x_{r_i}^i) S^{\mathcal{D}}(\mathcal{Y} \mid X^i = x_{r_i}^i), \qquad (6.53)$$

where $P^{\mathcal{D}}(X^i = x_{r_i}^i)$ is the probability that $X^i = x_{r_i}^i$ in $\mathcal{D}$, estimated via relative frequencies, and $S^{\mathcal{D}}(\mathcal{Y} \mid X^i = x_{r_i}^i)$ is the entropy of $\mathcal{Y}$ on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x_{r_i}^i, \quad \forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

At a leaf node, a label is predicted as relevant if, and only if, it is relevant for the majority of the instances at that leaf node. Formally, at a leaf node $\mathcal{L}$, let $n^{\mathcal{L}}(y_j)$ ($n^{\mathcal{L}}(\overline{y_j})$) denote the number of instances in $\mathcal{L}$ for y_j is relevant (irrelevant), $\quad \forall j = 1, 2, \ldots, n_L$. The predicted set of relevant labels at $\mathcal{L}$ is determined as follows:

$$h_{ML\_DT}^{\mathcal{L}} = \left\{ y_j \mid n^{\mathcal{L}}(y_j) > n^{\mathcal{L}}(\overline{y_j}), \quad j \in \{1, 2, \ldots, n_L\} \right\}. \qquad (6.54)$$

The posterior probability about the relevance of y_j at $\mathcal{L}$ is predicted via relative frequencies:

$$f^{\mathcal{L}}_{\mathrm{ML\_DT}}(y_j) = \frac{n^{\mathcal{L}}(y_j)}{N^{\mathcal{L}}}, \tag{6.55}$$

$N^{\mathcal{L}}$ being the total number of instances at $\mathcal{L}$.

Algorithm 9 summarizes the building procedure of a ML-DT.

Algorithm 9: Procedure to build a Multi-Label Decision Tree.

```
Procedure Build_ML-DT(Node $\mathcal{N}$)
Let $\mathcal{D}$ be the dataset associated with $\mathcal{N}$
if There are more attributes to insert then
    Select the attribute $X^i$ that reaches the maximum value of $IG^{\mathcal{D}}(\mathcal{Y}, X^i)$
    for $x^i_{r_i}$ possible value of $X^i$ do
        Make a node $\mathcal{N}_{r_i}$ child of $\mathcal{N}$
        Build_ML-DT($\mathcal{N}_{r_i}$)
else
    Make $\mathcal{N}$ a leaf node
    Assign a label set $h^{\mathcal{N}}_{\mathrm{ML\_DT}}$ to $\mathcal{N}$, computed through Equation (6.54)
    for $j = 1$ to $n_L$ do
        $f^{\mathcal{N}}_{\mathrm{ML\_DT}}(y_j) = \frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}}$
```

For classifying an instance via ML-DT, a path from the root node to a leaf one is made by using the attribute values of the instance. The predicted label set for the instance is the one assigned to such a terminal node. The same happens with the predicted posterior probabilities about the relevance of the labels for such an instance. The procedure to classify an instance with ML-DT is summarized in Algorithm 10.

Algorithm 10: Procedure to classify an instance with ML-DT.

```
Procedure Classify_ML-DT(ML-DT $\mathcal{T}$, instance with attribute vector
$\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$)
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the
   attribute values $x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}$.
2. Assign the predicted label set at $\mathcal{L}$, $h^{\mathcal{L}}$, to $h^{\mathrm{ML\_DT}}(\mathbf{x})$.
3. for $j = 1$ to $n_L$ do
       $f^{\mathrm{ML\_DT}}(\mathbf{x}, y_j) = f^{\mathcal{L}}_{\mathrm{ML\_DT}}(y_j)$
```

ML-DT can handle continuous attributes similarly to Decision Trees for traditional classification, considering binary splits and choosing the split point that produces the maximum gain of uncertainty-based information about the label set. Concerning missing values, when an instance has a missing value for an attribute, it can go down each branch hanging from the corresponding node with a weight equal to the proportion of instances at such a branch, as in Decision Trees for standard classification. In this case, it is necessary to adapt the entropy of each label, computed via Equation (6.51), for working with proportions of weights rather than proportions of instances.

Finally, ML-DT can use pruning processes based on the pruning processes of Decision Trees for traditional classification but considering the number of errors in all labels [56].

6.7 Multi-Label Nearest Neighbors

The Multi-Label Nearest Neighbor algorithm (ML-NN) [232], as the Nearest Neighbors algorithm for traditional classification, described in Section 4.3, is a lazy approach that does not carry out any training phase.

Suppose that it is wanted to classify an instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \text{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$. ML-NN computes the $num\_neighbors$-nearest neighbors of the instance by using a distance function on the attribute space. For each label $y_j \in \mathcal{Y}$, let $\mathcal{K}_j(\mathbf{x})$ be the number of neighbors of the instance (among the $num\_neighbors$-nearest ones) that have associated y_j.

Let $P^{ML-NN}(y_j)$ ($P^{ML-NN}(\overline{y_j})$) denote the prior probability that y_j is relevant (irrelevant) for an instance, $\quad \forall j = 1, 2, \ldots, n_L$. Also, for each label $y_j \in \mathcal{Y}$, let $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$ ($P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$) denote the probability that an instance has $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j conditioned on y_j is relevant (irrelevant) for such an instance.

ML-NN uses a Maximum a Posteriori Principle (MAP) to predict whether each label $y_j \in \mathcal{Y}$ is relevant for the instance. According to such a principle, y_j is predicted as relevant for the instance if, and only if,

$$P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) > P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}). \tag{6.56}$$

Let N_{tr} be the total number of training instances. For each $j = 1, 2, \ldots, n_L$, let $n_{tr}(y_j)$ ($n_{tr}(\overline{y_j})$) be the number of training instances for which y_j is relevant (irrelevant). Let $\delta_j(\mathcal{K}_j(\mathbf{x}))$ denote the number of training instances that have associated y_j and have $\mathcal{K}_j(\mathbf{x})$ neighbors for which y_j is relevant. Likewise, let $\overline{\delta_j}(\mathcal{K}_j(\mathbf{x}))$ denote the number of training instances for which y_j is irrelevant and have $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j. ML-NN estimates the mentioned probabilities that appear in Equation (6.56) via relative frequencies with Laplacian correction:

$$P^{ML-NN}(y_j) = \frac{n_{tr}(y_j)+1}{N_{tr}+2}, \quad P^{ML-NN}(\overline{y_j}) = \frac{n_{tr}(\overline{y_j})+1}{N_{tr}+2}, \tag{6.57}$$

$$\begin{aligned} P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) &= \frac{\delta_j(\mathcal{K}_j(\mathbf{x}))+1}{n_{tr}(y_j)+num\_neighbors+1}, \\ P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}) &= \frac{\overline{\delta_j}(\mathcal{K}_j(\mathbf{x}))+1}{n_{tr}(\overline{y_j})+num\_neighbors+1}, \end{aligned} \tag{6.58}$$

Hence, given an instance with attribute vector $\mathbf{x} = \left(x_{r_1}^1, x_{r_2}^2, \dots, x_{r_d}^d\right)$, where $x_{r_i}^i \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \dots, d$, the set of labels predicted by ML-NN as relevant for such an instance is determined in the following way:

$$\begin{aligned} h^{ML\_NN}(\mathbf{x}) = \{y_j \mid P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) > \\ P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}), \quad j \in \{1, 2, \dots, n_L\}\}, \end{aligned} \tag{6.59}$$

where $P^{ML-NN}(y_j)$ and $P^{ML-NN}(\overline{y_j})$ are computed by means of Equation (6.57), and $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$ and $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$ via Equation (6.58).

For each label $y_j \in \mathcal{Y}$, the predicted posterior probability that the instance has associated y_j is determined by normalizing the probability that y_j is relevant for the instance estimated via MAP:

$$f^{ML\_NN}(\mathbf{x}, y_j) = \frac{P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)}{P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) + P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})}. \tag{6.60}$$

6.7.1 Modifications of ML-KNN based on classical information theory

ML-NN was the first lazy approach to MLC. Many lazy MLC algorithms have been developed since them. Most of them are based on classical probability theory. Among such methods, we can mention the following ones:

- The Binary Relevance Nearest Neighbors algorithm (BR-NN) was proposed in [194]. To predict whether a label is relevant for an instance, BR-NN uses a majority vote in the neighborhood of the instance[3].

[3] BR-NN is, indeed, a problem transformation method. Specifically, it is the Binary Relevance method using the NN algorithm for traditional classification as the base classifier.

Formally, suppose that it is wanted to classify an instance whose attribute vector is $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$. As ML-NN, BR-NN determines the $num\_neighbors$-nearest neighbors of the instance via a distance function on the attribute space.

For each label $y_j \in \mathcal{Y}$, let $\mathcal{K}_j(\mathbf{x})$ ($\overline{\mathcal{K}_j}(\mathbf{x})$) denote the number of neighbors of the instance for which y_j is relevant (irrelevant). The set of labels predicted by BR-NN as relevant for the instance is the one given by:

$$h^{BR-NN}(\mathbf{x}) = \left\{y_j \mid \mathcal{K}_j(\mathbf{x}) > \overline{\mathcal{K}_j}(\mathbf{x}), \quad j \in \{1, 2, \ldots, n_L\}\right\}. \tag{6.61}$$

The posterior probability that the instance has associated a label is predicted through relative frequencies with Laplacian correction in the neighborhood:

$$f^{BR-NN}(\mathbf{x}, y_j) = \frac{\mathcal{K}_j(\mathbf{x}) + 1}{num\_neighbors + 2}, \quad \forall j = 1, 2, \ldots, n_L. \tag{6.62}$$

Two extensions of BR-NN were proposed in [194]. The first one is known as BR-NN-α. It predicts, for an instance with attribute vector $\mathbf{x} = \left(x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d}\right)$, where $x^i_{r_i} \in \mathrm{Dom}(X^i) \quad \forall i = 1, 2, \ldots, d$, the label with the highest predicted posterior probability as relevant when $h^{BR-NN}(\mathbf{x}) = \emptyset$. The second extension, called BR-NN-β, always predicts the β top-ranked labels as relevant, β being the average number of relevant labels on the neighboring instances.

- A new version of the ML-NN algorithm called Dependent Multi-Label Nearest Neighbors (DML-NN) was proposed in [226]. As ML-NN, it predicts the set of relevant labels for an instance by utilizing the MAP principle. Nonetheless, in order to predict whether a label is relevant for an instance, DML-NN also takes into account the neighboring instances for which each one of the rest of the labels is relevant, whereas ML-NN only considers the number of neighboring instances that have associated the label to predict. In this way, DML-NN aims to exploit correlations between labels.

- In [54], an MLC algorithm that combines instance-based learning with logistic regression (IBLR-ML) was proposed. It transforms the training dataset by creating features using label information and considers a logistic regression classifier per label. Due to these transformations and the logistic regression classifiers, IBLR-ML is computationally expensive.

- The Multi-Label Classification Weighted Nearest Neighbors algorithm (MLCW-NN) was developed in [221]. It consists of an instance-weighted version of ML-NN in which, among the $num\_neighbors$-nearest neighbors, the nearest instances influence more than the most distant ones. MLCW-NN uses a quadratic programming method to estimate the weight of each neighboring instance. Thereby, the computational cost of MLCW-NN is also notably high.

Part III

CONTRIBUTIONS OF OUR THESIS

7 ANALYSIS OF IMPRECISE PROBABILITY THEORIES AND MODELS

7.1 Introduction

The use of classical probability theory is the standard way of representing the probabilistic knowledge about a discrete variable or finite set. This representation may be suitable in many cases. However, as explained before, in some situations, a single probability distribution might not be appropriate since the available information is not sufficient. For this reason, several mathematical theories and models based on imprecise probabilities have been developed in the literature. We described most of them in Chapter 2.

Regarding imprecise probability theories, one of the most general is the one based on credal sets. In fact, in most imprecise probability theories, the available information can be represented via a credal set. Nonetheless, as these theories have specific mathematical properties, some theories are more suitable than others in specific situations.

On the one hand, Evidence theory (ET) has been commonly used in the literature to deal with uncertainty-based information. It has been successfully applied to several domains such as *statistical classification* [78], *target identification* [41], *medical diagnosis* [34], and *face recognition* [120]. Moreover, this theory has been frequently employed for the fusion of information provided by different sources, which is very important for decision making [30, 53, 165]. In ET, the available information can1 be represented by a *belief function*.

On the other hand, reachable probability intervals are easy to understand and manage. They have high expressive power and can be efficiently computed. For these reasons, reachable probability intervals have been frequently employed in practical applications such as classification. Examples of this point can be found in Chapters 4 and 5.

As we highlighted in Section 2.2.8, ET does not generalize reachable probability intervals, and the converse is also not satisfied (See Figure 2.1). Abellán [2] demonstrated that a reachable set of probability intervals is not necessarily associated with a belief function, and a belief function cannot always be represented by means of a reachable set of probability intervals. A characterization of reachable probability intervals that can be represented via belief functions was given in [135]. Nevertheless, we show with a counterexample that the condition given in that study is not sufficient because the belief function found under that condition does not always represent the same information as the corresponding set of probability intervals.

In this chapter, as a novelty, we study the necessary and sufficient conditions under which a reachable set of probability intervals corresponds to a belief function. Also, we analyze the properties that a belief function must satisfy to be representable via a reachable set of probability intervals. In this way, we describe the credal sets that belong to both belief functions and reachable probability intervals. The credal sets corresponding to the Imprecise Dirichlet Model (IDM) are illustrative examples of this type of credal set [2].

Concerning imprecise probability models, as shown in Section 2.3.1, the IDM is based on reachable probability intervals and has been frequently used in the literature. It satisfies some principles that were claimed to be desirable for inference such as the *Representation Invariance Principle* (RIP). However, as pointed out before, the IDM assumes previous knowledge about the data via a parameter. In classification, small changes in the IDM parameter lead to important variations in the results, and each classification dataset has associated with it an optimal value of the IDM parameter [18, 150].

The Non-Parametric Predictive Inference Model for Multinomial data (NPI-M) was developed in [58, 59]. It is a non-parametric approach that does not make prior assumptions about the data before observing them. In Section 2.3.2, we showed that, in many situations, inferences with the NPI-M yield intuitively more coherent results than inferences with the IDM. The NPI-M has been successfully used in practical applications during the last years, such as credit scoring [60], European option pricing [111], or extraction of knowledge in traffic accident databases [9]. The NPI-M has equivalent performance to the IDM with the best selection of the parameter when both models are utilized in classification [6].

Despite the previous points, it should be noted that the set of probability distributions compatible with the NPI-M is not convex. Thus, when the NPI-M is employed, it is needed to handle difficult constraints, as highlighted in Section 2.3.2. For this reason, the Approximate Non-Parametric Predictive Inference Model for Multinomial data (A-NPI-M) was proposed in [5]. It corresponds to the convex hull of the set of probability distributions compatible with the NPI-M. The A-NPI-M belongs to reachable probability intervals theory and avoids many difficult constraints of the exact model. In classification, the NPI-M and the A-NPI-M have obtained equivalent results [6].

As a novelty, in this chapter, we analyze the properties of credal sets associated with the A-NPI-M, comparing them with the properties of IDM credal sets. One of the most remarkable properties is that A-NPI-M credal sets are not always representable via belief functions, unlike IDM credal sets. We show that the A-NPI-M is a more complex model than the IDM. Nevertheless, we must remark that the latter model assumes previous knowledge about the data via a parameter, unlike the former.

This chapter is organized in the following way: Credal sets representable through reachable sets of probability intervals and belief functions are characterized in Section 7.2. In Section 7.3, we analyze the properties of credal sets associated with the A-NPI-M. Concluding remarks are given in Section 7.4.

7.2 Reachable probability intervals and belief functions

Let $X = \{x_1, \dots, x_t\}$ be a finite set of possible alternatives[1].

In Example 2.2.2, we have illustrated a case in which the Möbius inverse of the natural extension of a reachable set of probability intervals on X is not non-negative. Also, Example 2.2.1 has shown a case in which the credal set associated with a belief function on X does not coincide with the credal set compatible with the corresponding set of belief intervals for singletons. Therefore, a reachable set of probability intervals on X cannot always be represented by means of a belief function on X, and a belief function on X is not always representable via a reachable set of probability intervals on X.

7.2.1 Reachable probability intervals representable by belief functions

Let $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \dots, t\}$ be a reachable set of probability intervals on X.

According to the results proved in [135], $\mathcal{I}$ can be represented via a belief function if, and only if,

$$\sum_{i=1}^{t} l_i + \sum_{i=1}^{t} u_i \geqslant 2. \tag{7.1}$$

Indeed, this condition is necessary, but it is not sufficient, as the following example shows:

[1] or, alternatively, a discrete variable whose set of possible values is $\{x_1, \dots, x_t\}$.

Example 7.2.1 *Let $X = \{x_1, x_2, x_3, x_4\}$ be a finite set. Let us consider the following reachable set of probability intervals on X:*

$$\mathcal{I} = \{[0, 0.5]\,;\,[0, 0.5]\,;\,[0, 0.6]\,;\,[0, 0.7]\}.$$

Let $\underline{P}$ denote the natural extension of $\mathcal{I}$ and m its corresponding Möbius inverse. We have that:

$$\sum_{i=1}^{4} l_i + \sum_{i=1}^{4} u_i = 2.3 > 2.$$

Within this example, we use Proposition 2.2.7 to calculate $\underline{P}$.

$$m(\{x_i\}) = \underline{P}(\{x_i\}) = \max\left(l_i, 1 - \sum_{j=1, j\neq i}^{4} u_j\right) = l_i = 0, \quad \forall i = 1,2,3,4.$$

$$\underline{P}\left(\{x_i, x_j\}\right) = \max\left(l_i + l_j, 1 - u_k - u_l\right) = 0, \quad \forall 1 \leqslant i < j \leqslant 4,$$

where $1 \leqslant k < l \leqslant 4$, with $\{i,j\} \cap \{k,l\} = \emptyset$.

$$m\left(\{x_i, x_j\}\right) = \underline{P}\left(\{x_i, x_j\}\right) - \underline{P}(\{x_i\}) - \underline{P}\left(\{x_j\}\right) = 0, \quad \forall 1 \leqslant i < j \leqslant 4.$$

$$\begin{aligned}
\underline{P}(\{x_1, x_2, x_3\}) &= \max(l_1 + l_2 + l_3, 1 - u_4) = 0.3,\\
\underline{P}(\{x_1, x_2, x_4\}) &= \max(l_1 + l_2 + l_4, 1 - u_3) = 0.4,\\
\underline{P}(\{x_1, x_3, x_4\}) &= \max(l_1 + l_3 + l_4, 1 - u_2) = 0.5,\\
\underline{P}(\{x_2, x_3, x_4\}) &= \max(l_2 + l_3 + l_4, 1 - u_1) = 0.5.
\end{aligned}$$

Since $m(A) = 0 \quad \forall A \subseteq X$ such that $|A| \leqslant 2$, it holds that

$$\begin{aligned}
m(\{x_1, x_2, x_3\}) &= \underline{P}(\{x_1, x_2, x_3\}) = 0.3,\\
m(\{x_1, x_2, x_4\}) &= \underline{P}(\{x_1, x_2, x_4\}) = 0.4,\\
m(\{x_1, x_3, x_4\}) &= \underline{P}(\{x_1, x_3, x_4\}) = 0.5,\\
m(\{x_2, x_3, x_4\}) &= \underline{P}(\{x_2, x_3, x_4\}) = 0.5.
\end{aligned}$$

$$\begin{aligned}
m(X) &= 1 - \sum_{A \subset X} m(A)\\
&= 1 - m(\{x_1, x_2, x_3\}) - m(\{x_1, x_2, x_4\})\\
&- m(\{x_1, x_3, x_4\}) - m(\{x_2, x_3, x_4\})\\
&= 1 - 0.3 - 0.4 - 0.5 - 0.5 = -0.7 < 0.
\end{aligned}$$

Consequently, even though $\sum_{i=1}^{4} l_i + \sum_{i=1}^{4} u_i \geqslant 2$, the Möbius inverse m is not non-negative and, thus, $\mathcal{I}$ cannot be represented via a belief function.

In [135], it was shown that, if the inequality given in Equation (7.1) is satisfied, then it is possible to find a BPA associated with $\mathcal{I}$. However, such a BPA does not always coincide with the Möbius inverse corresponding to the natural extension of $\mathcal{I}$ and, consequently, it does not always represent the same information.

Let $\underline{P}$ denote the natural extension of $\mathcal{I}$ and m its associated Möbius inverse. In this section, we aim to analyze in which cases $\underline{P}$ is a belief function. For this purpose, we need to study the properties that have to be satisfied by the extreme values of the intervals for m to be non-negative.

The Möbius inverse for singletons is clearly greater or equal than 0:

$$m\left(\{x_i\}\right) = \underline{P}\left(\{x_i\}\right) = l_i \geqslant 0, \quad \forall i = 1, 2, \ldots, t.$$

We use the following lemma for posterior results:

Lemma 7.2.1

$$\sum_{i=0}^{n} (-1)^i \binom{n}{i} = (1-1)^n = 0, \quad \forall n \in \mathbb{N},\, n \geqslant 1.$$

We present a property that, if it is satisfied by a set with cardinality greater or equal than 2, then the Möbius inverse for that set is equal to 0. The following proposition is useful for it:

Proposition 7.2.1 *$\forall A \subseteq X$ such that $|A| \geqslant 2$, it holds that:*

$$\sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i = 0.$$

Proof:

In order to determine the number of subsets of A with a certain cardinality that contain x_i, we think in the following way: for a subset $B \subseteq A$ containing x_i, there can be j elements of A that do not belong to B, with $0 \leqslant j \leqslant |A| - 1$. The j elements can be chosen in $\binom{|A|-1}{j}$ different ways. Hence,

$$\begin{aligned}\sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i &= \sum_{x_i \in A} l_i \times \left(\sum_{j=0}^{|A|-1} (-1)^j \binom{|A|-1}{j} \right) \\ &= \sum_{x_i \in A} l_i \times 0 = 0,\end{aligned}$$

where we have used Lemma 7.2.1 in the penultimate equality taking into account that $|A| - 1 \geqslant 1$.

□

Now, we have the following result:

Proposition 7.2.2 *If $A \subseteq X$, with $|A| \geqslant 2$, satisfies that*

$$\sum_{x_i \in A} l_i \geqslant 1 - \sum_{x_i \notin A} u_i,$$

then $\underline{P}(B) = \sum_{x_i \in B} l_i \quad \forall B \subseteq A$ *and* $m(A) = 0$.

Proof:

Under our hypothesis, if $B \subseteq A$:

$$\begin{aligned} \sum_{x_i \in B} l_i &= \sum_{x_i \in A} l_i - \sum_{x_i \in A \setminus B} l_i \geqslant 1 - \sum_{x_i \notin A} u_i - \sum_{x_i \in A \setminus B} l_i \\ &\geqslant 1 - \sum_{x_i \notin A} u_i - \sum_{x_i \in A \setminus B} u_i \\ &= 1 - \sum_{x_i \notin B} u_i. \end{aligned}$$

In consequence,

$$\underline{P}(B) = \sum_{x_i \in B} l_i, \quad \forall B \subseteq A.$$

Proposition 7.2.1 allows concluding that:

$$\begin{aligned} m(A) &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} \underline{P}(B) \\ &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i = 0. \end{aligned}$$

□

The inverse is not satisfied. A set with cardinality greater or equal than 2 that has a Möbius inverse equal to 0 might not verify the property of the previous proposition, as shown in the following example.

Example 7.2.2 *Suppose that we have a finite set $X = \{x_1, x_2, x_3\}$ and the following set of probability intervals on X:*

$$\{[l_1, u_1]; [l_2, u_2]; [l_3, u_3]\} = \{[0, 1]; [0, 0.1]; [0, 0.9]\}$$

.

It is easy to check that this set of probability intervals is reachable. Let $\underline{P}$ be the natural extension of $\mathcal{I}$ and m its corresponding Möbius inverse. As in the previous examples, we calculate $\underline{P}$ via Proposition 2.2.7.

For singletons, we have that:

$$m(\{x_i\}) = \underline{P}(\{x_i\}) = l_i = 0, \quad \forall i = 1, 2, 3.$$

For sets with cardinality equal to 2:

$$\underline{P}(\{x_1, x_2\}) = \max(l_1 + l_2, 1 - u_3) = \max(0, 0.1) = 0.1,$$

$$m(\{x_1, x_2\}) = \underline{P}(\{x_1, x_2\}) - \underline{P}(\{x_1\}) - \underline{P}(\{x_2\}) = 0.1,$$

$$\underline{P}(\{x_1, x_3\}) = \max(l_1 + l_3, 1 - u_2) = \max(0, 0.9) = 0.9,$$

$$m(\{x_1, x_3\}) = \underline{P}(\{x_1, x_3\}) - \underline{P}(\{x_1\}) - \underline{P}(\{x_3\}) = 0.9,$$

$$\underline{P}(\{x_2, x_3\}) = \max(l_2 + l_3, 1 - u_1) = \max(0, 0) = 0,$$

$$m(\{x_2, x_3\}) = \underline{P}(\{x_2, x_3\}) - \underline{P}(\{x_2\}) - \underline{P}(\{x_3\}) = 0.$$

Now,

$$\begin{aligned} m(\{x_1, x_2, x_3\}) = 1 - m(\{x_1\}) - m(\{x_2\}) - m(\{x_3\}) - \\ m(\{x_1, x_2\}) - m(\{x_1, x_3\}) - m(\{x_2, x_3\}) \\ = 1 - 0 - 0 - 0 - 0.1 - 0.9 - 0 = 0, \end{aligned}$$

and $l_1 + l_2 + l_3 = 0 < 1$.

So, even though $m(X) = 0$, *it does not hold that* $\sum_{x_i \in X} l_i \geqslant 1 - \sum_{x_i \notin X} u_i$.

As a consequence of Proposition 7.2.2, we have the three following results:

Corollary 7.2.1 *If* A *is a subset of smallest cardinality that satisfies*

$$\sum_{x_i \in A} l_i < 1 - \sum_{x_i \notin A} u_i,$$

then $m(A) = 1 - \sum_{x_i \in A} l_i - \sum_{x_i \notin A} u_i > 0$.

Proof:

Since $\mathcal{I}$ is proper, it holds that $0 \geqslant 1 - \sum_{x_i \in X} u_i$ and, consequently, A cannot be equal to the empty set. If $|A| = 1$, due to the reachability condition, then, it is not possible that $\sum_{x_i \in A} l_i < 1 - \sum_{x_i \notin A} u_i$. Thus, $|A| \geqslant 2$.

By hyphotesis,

$$\underline{P}(A) = 1 - \sum_{x_i \notin A} u_i,$$

and

$$\sum_{x_i \in B} l_i \geqslant 1 - \sum_{x_i \notin B} u_i, \quad \forall B \subset A.$$

From Proposition 7.2.2, it follows that

$$\underline{P}\,(B) = \sum_{x_i \in B} l_i, \quad \forall B \subset A : |B| \geqslant 2.$$

Furthermore, due to the reachability condition, $\underline{P}\,(\{x_i\}) = l_i, \quad \forall x_i \in B$. Thereby,

$$\underline{P}\,(B) = \sum_{x_i \in B} l_i \quad \forall B \subset A.$$

Hence,

$$\begin{aligned} m\,(A) &= \sum_{B \subseteq A} (-1)^{|A \setminus B|}\, \underline{P}\,(B) = \underline{P}\,(A) + \sum_{B | B \subset A} (-1)^{|A \setminus B|}\, \underline{P}\,(B) \\ &= 1 - \sum_{x_i \notin A} u_i + \sum_{B | B \subset A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i. \end{aligned}$$

Now, due to Proposition 7.2.1, it is satisfied that:

$$\begin{aligned} m\,(A) = m\,(A) - 0 = m\,(A) &- \sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i \\ &= 1 - \sum_{x_i \notin A} u_i + \sum_{B | B \subset A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i - \sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_i \in B} l_i \\ &= 1 - \sum_{x_i \notin A} u_i - \sum_{x_i \in A} l_i > 0. \end{aligned}$$

□

Corollary 7.2.2 *If for each* $A \subseteq X$ *such that* $|A| = t'$, *with* $1 \leqslant t' < t$, *it is satisfied that*

$$\sum_{x_i \in A} l_i \geqslant 1 - \sum_{x_i \notin A} u_i,$$

then $m(B) \geqslant 0 \quad \forall B \subseteq X$ *such that* $|B| = t' + 1$.

Proof: Under our hypothesis, for each $B \subseteq X$ such that $|B| = t' + 1$, it holds that $|B| \geqslant 2$, and there are two possibilities:

1. $\sum_{x_i \in B} l_i \geqslant 1 - \sum_{x_i \notin B} u_i$.
   In this case, Proposition 7.2.2 allows us to deduce that $m(B) = 0$.

2. $\sum_{x_i \in B} l_i < 1 - \sum_{x_i \notin B} u_i$.
   Then, B is a subset with smallest cardinality that satisfies this condition. From Corollary 7.2.1, it follows that $m(B) > 0$.

□

Corollary 7.2.3 *For each subset $A \subseteq X$ such that $|A| = 2$, it holds that $m(A) \geqslant 0$.*

Proof: It immediately follows from Corollary 7.2.2 and the reachability condition. Moreover, this result is a direct consequence of the fact that reachable probability intervals are particular cases of Choquet capacities of order 2. □

Due to the proper condition, it is satisfied that $\sum_{i=1}^{t} l_i \leqslant 1$. It is easy to observe that, if $\sum_{i=1}^{t} l_i = 1$, then $l_i = u_i \quad \forall i = 1, 2, \ldots, t$ because $\mathcal{I}$ is reachable. In this case, there is a single probability distribution compatible with $\mathcal{I}$, and it is obvious that $\mathcal{I}$ can be represented via a belief function, which coincides with the unique probability distribution consistent with $\mathcal{I}$. For this reason, hereon, we assume that $\sum_{i=1}^{t} l_i < 1$.

In order to determine the conditions under which m is non-negative, we consider the following set of probability intervals on X:

$$\mathcal{I}' = \left\{ [l'_i, u'_i],\, l'_i = 0,\, u'_i = \frac{u_i - l_i}{1 - L}, \quad i = 1, 2, \ldots, t \right\}, \tag{7.2}$$

where $L = \sum_{i=1}^{t} l_i < 1$.

This set of probability intervals is reachable, as the following result shows:

Proposition 7.2.3 *$\mathcal{I}'$ is a reachable set of probability intervals.*

Proof: As $\mathcal{I}$ is a reachable set of probability intervals, due to Proposition 2.2.6, it holds that:

1.

$$u_i + \sum_{j=1, j\neq i}^{t} l_j \leqslant 1 \Rightarrow u_i \leqslant 1 - L + l_i \Rightarrow \frac{u_i - l_i}{1 - L} \leqslant 1 \Rightarrow \sum_{j=1, j\neq i}^{t} 0 + u'_i \leqslant 1$$
$$\Rightarrow \sum_{j=1, j\neq i}^{t} l'_j + u'_i \leqslant 1, \quad \forall i = 1, 2, \ldots, t.$$

2.

$$
\begin{aligned}
l_i + \sum_{j=1,j\neq i}^{t} u_j \geqslant 1 &\Rightarrow \sum_{j=1}^{t} l_j - \sum_{j=1,j\neq i}^{t} l_j + \sum_{j=1,j\neq i}^{t} u_j \geqslant 1 \\
&\Rightarrow \sum_{j=1,j\neq i}^{t} u_j - \sum_{j=1,j\neq i}^{t} l_j \geqslant 1 - L \\
&\Rightarrow \sum_{j=1,j\neq i}^{t} \frac{u_j - l_j}{1-L} \geqslant 1 \Rightarrow \sum_{j=1,j\neq i}^{t} u'_j \geqslant 1 \\
&\Rightarrow \sum_{j=1,j\neq i}^{t} u'_j + l'_i = \sum_{j=1,j\neq i}^{t} u'_j + 0 \geqslant 1, \quad \forall i = 1,2,\dots,t,
\end{aligned}
$$

and Proposition 2.2.6 let us conclude that $\mathcal{I}'$ is reachable.

□

Let $\underline{P}_1$ denote the natural extension of $\mathcal{I}'$ and m_1 its associated Möbius inverse:

$$\underline{P}_1(A) = \max\left\{\sum_{x_i\in A} l'_i, 1 - \sum_{x_i\notin A} u'_i\right\} = \max\left\{0, 1 - \sum_{x_i\notin A} u'_i\right\},$$

$$m_1(A) = \sum_{B\subseteq A} (-1)^{|A\setminus B|}\, \underline{P}_1(B), \quad \forall A \subseteq X.$$

According to the following result, $\mathcal{I}$ can be represented by a belief function if, and only if, $\underline{P}_1$ is a belief function.

Theorem 7.2.1 *m is non-negative if, and only if, m_1 is non-negative.*

Proof: Since m is the Möbius inverse associated with $\underline{P}$ (the natural extension of $\mathcal{I}$), and m_1 is the Möbius inverse corresponding to $\underline{P}_1$:

$$
\begin{aligned}
m(A) &= \sum_{B\subseteq A} (-1)^{|A\setminus B|}\underline{P}_1(B) \\
&= \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(\sum_{x_i\in B} l_i, 1 - \sum_{x_i\notin B} u_i\right), \quad \forall A \subseteq X.
\end{aligned}
$$

$$\begin{aligned} m_1(A) &= \sum_{B\subseteq A} (-1)^{|A\setminus B|}\underline{P}(B) = \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(0, 1-\sum_{x_i\notin B} u_i'\right) \\ &= \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(0, 1-\sum_{x_i\notin B} \frac{u_i-l_i}{1-L}\right) \\ &= \frac{1}{1-L}\times \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(0, 1-L+\sum_{x_i\notin B} l_i-\sum_{x_i\notin B} u_i\right) \\ &= \frac{1}{1-L}\times \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(0, 1-\sum_{x_i\in B} l_i-\sum_{x_i\notin B} u_i\right), \quad \forall A\subseteq X. \end{aligned}$$

Now, due to Proposition 7.2.1, it holds that, for each $A \subseteq X$ with $|A| \geqslant 2$:

$$\begin{aligned} m(A) &= m(A) - 0 \\ &= \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(\sum_{x_i\in B} l_i, 1-\sum_{x_i\notin B} u_i\right) - \sum_{B\subseteq A} (-1)^{|A\setminus B|} \sum_{x_i\in B} l_i \\ &= \sum_{B\subseteq A} (-1)^{|A\setminus B|} \max\left(0, 1-\sum_{x_i\in B} l_i-\sum_{x_i\notin B} u_i\right), \end{aligned}$$

and it is immediate that $m(A) = (1-L)m_1(A), \quad \forall A \subseteq X$ with $|A| \geqslant 2$.

Furthermore, we must remark that, for singletons, m is non-negative and m_1 is equal to 0. Also, it is obvious that $m(\emptyset) = m_1(\emptyset) = 0$.

Therefore, it can be concluded that the Möbius inverse corresponding to the natural extension of $\mathcal{I}$ is non-negative if, and only if, the one associated with the natural extension of $\mathcal{I}'$ is non-negative. □

Hence, we will focus on studying when m' is non-negative.

We partition the set X as follows: $X = X_1 \cup X_2$, where:

$$X_1 = \left\{x_i \in X \mid u_i' = 1, \quad 1 \leqslant i \leqslant t\right\}, \quad X_2 = \left\{x_i \in X \mid u_i' < 1, \quad 1 \leqslant i \leqslant t\right\}.$$

Clearly,

$$X_1 = \left\{x_i \in X \mid u_i + \sum_{j=1, j\neq i}^{t} l_j = 1, \quad 1 \leqslant i \leqslant t\right\},$$

$$X_2 = \left\{x_i \in X \mid u_i + \sum_{j=1, j\neq i}^{t} l_j < 1, \quad 1 \leqslant i \leqslant t\right\}.$$

As the following results show, the value of m_1 for the subsets of X that do not contain X_1 is equal to 0.

Proposition 7.2.4 *If* $A \subseteq X$ *satisfies that* $X_1 \not\subseteq A$*, then* $\underline{P}_1(A) = 0$*.*

Proof: Under our hypothesis:

$$\underline{P}_1(A) = \max\left\{\sum_{x_i \in A} 0, 1 - \sum_{x_i \notin A} u_i'\right\}$$
$$= \max\left\{0, 1 - \sum_{x_i \in X_1 \setminus A} u_i' - \sum_{x_i \notin A \cup X_1} u_i'\right\} = 0,$$

since $X_1 \setminus A \neq \emptyset$ and $u_i' = 1 \quad \forall x_i \in X_1$.

□

Corollary 7.2.4 *If* $A \subseteq X$ *verifies that* $X_1 \not\subseteq A$*, then* $\underline{P}_1(B) = 0 \quad \forall B \subseteq A$*.*

Proof: It is sufficient to observe that, if $X_1 \not\subseteq A$ and $B \subseteq A$, then $X_1 \not\subseteq B$ and apply Proposition 7.2.4. □

Corollary 7.2.5 *If* $A \subseteq X$ *satisfies that* $X_1 \not\subseteq A$*, then* $m_1(A) = 0$*.*

Now, we distinguish two cases:

- **Case 1**: $\sum_{x_i \in X_2} u_i' \leqslant 1$.
- **Case 2**: $\sum_{x_i \in X_2} u_i' > 1$.

We show that, in Case 1, the given reachable set of probability intervals can be represented by a belief function.

Proposition 7.2.5 *If* $\sum_{x_i \in X_2} u_i' \leqslant 1$*, then* m_1 *is non-negative.*

Proof: According to Corollary 7.2.5, $m_1(A) = 0 \quad \forall A$ such that $X_1 \not\subseteq A$. Thereby, $m_1(A) \neq 0$ can only happen if $X_1 \subseteq A$. In consequence, we only need to check the sets A of the form $A = X_1 \cup B$, with $B \subseteq X_2$, because these are the only sets satisfying $X_1 \subseteq A$.

$$m_1(X_1) = \underline{P}_1(X_1) - \sum_{A \subset X_1} m_1(A) = \underline{P}_1(X_1) =$$
$$1 - \sum_{x_i \notin X_1} u_i' = 1 - \sum_{x_i \in X_2} u_i' \geqslant 0.$$

$$1-\sum_{x_i\in X_2} u_i' \geqslant 0 \Rightarrow 1-\sum_{x_i\notin X_1} u_i' \geqslant 0 \Rightarrow 1-\sum_{x_i\notin X_1\cup\{x_j\}} u_i' \geqslant 0$$

$$\Rightarrow \underline{P}_1(X_1\cup\{x_j\}) = 1-\sum_{x_i\notin X_1\cup\{x_j\}} u_i', \quad \forall x_j\in X_2.$$

$$\begin{aligned} m_1(X_1\cup\{x_j\}) &= \underline{P}_1(X_1\cup\{x_j\}) - m_1(X_1) \\ &= 1-\sum_{x_i\notin X_1\cup\{x_j\}} u_i' - (1-\sum_{x_i\in X_2} u_i') = \sum_{x_i\in X_2} u_i' - \sum_{x_i\in X_2\setminus\{x_j\}} u_i' \\ &= u_j' \geqslant 0, \quad \forall x_j\in X_2. \end{aligned}$$

We prove that $m_1(X_1\cup A)=0 \quad \forall A\subseteq X_2$ such that $|A|\geqslant 2$ by induction on $|A|$.

For $|A|=2$, i.e $A=\{x_j,x_k\}$, with $x_j,x_k\in X_2$, we have that:

$$\begin{aligned} m_1(X_1\cup A) &= m_1(X_1\cup\{x_j,x_k\}) = \\ & \underline{P}_1(X_1\cup\{x_j,x_k\}) - m_1(X_1\cup\{x_j\}) - m_1(X_1\cup\{x_k\}) - m_1(X_1) \\ &= 1-\sum_{x_i\notin X_1\cup\{x_j,x_k\}} u_i' - u_j' - u_k' - (1-\sum_{x_i\notin X_1} u_i') \\ &= \sum_{x_i\notin X_1} u_i' - \sum_{x_i\notin X_1\cup\{x_j,x_k\}} u_i' - u_j' - u_k' \\ &= u_j' + u_k' - u_j' - u_k' = 0. \end{aligned}$$

Suppose that it holds that $m_1(X_1\cup A)=0 \quad \forall A\subseteq X_2$ such that $2\leqslant|A|\leqslant t'$, for some $t'<|X_2|$. Let us assume that $B\subseteq X_2$ with $|B|=t'+1$. Then:

$$m_1(X_1\cup B) = \underline{P}_1(X_1\cup B) - \sum_{C\subset B} m_1(X_1\cup C).$$

By hypothesis of induction:

$$\begin{aligned} m_1(X_1\cup B) &= \underline{P}_1(X_1\cup B) - \sum_{C\subset B} m_1(X_1\cup C) \\ &= \underline{P}_1(X_1\cup B) - \sum_{x_i\in B} m_1(X_1\cup\{x_i\}) - m_1(X_1) \\ &= 1-\sum_{x_i\notin X_1\cup B} u_i' - \sum_{x_i\in B} u_i' - \left(1-\sum_{x_i\notin X_1} u_i'\right) \\ &= \sum_{x_i\notin X_1} u_i' - \sum_{x_i\notin X_1\cup B} u_i' - \sum_{x_i\in B} u_i' \\ &= \sum_{x_i\in B} u_i' - \sum_{x_i\in B} u_i' = 0. \end{aligned}$$

□

The following result is an immediate consequence of the previous proposition.

Corollary 7.2.6 *If* $\sum_{x_i \in X_1} l_i \leqslant 1 - \sum_{x_i \in X_2} u_i$*, then* $\mathcal{I}$ *can be represented by a belief function.*

Proof: It is enough to observe that

$$\begin{aligned}\sum_{x_i \in X_2} u_i' \leqslant 1 \Leftrightarrow \sum_{x_i \in X_2} \frac{u_i - l_i}{1 - L} \leqslant 1 \Leftrightarrow \sum_{x_i \in X_2} u_i - l_i \leqslant 1 - L \\ \Leftrightarrow \sum_{x_i \in X_2} u_i \leqslant 1 - \sum_{x_i \notin X_2} l_i = 1 - \sum_{x_i \in X_1} l_i \\ \Leftrightarrow \sum_{x_i \in X_1} l_i \leqslant 1 - \sum_{x_i \in X_2} u_i.\end{aligned}$$

and apply the previous proposition and Theorem 7.2.1. □

We study **Case 2**: $\sum_{x_i \in X_2} u_i' > 1$.

Let $\mathcal{I}_2$ be the set of probability intervals on the subset of X_2 composed of those elements for which the upper probability is not equal to 0:

$$\mathcal{I}_2 = \left\{ [0, u_i'] \mid x_i \in X_2 \wedge u_i' > 0 \right\}. \qquad (7.3)$$

Let $\underline{P}_2$ denote the natural extension of $\mathcal{I}_2$, $\overline{P}_2$ its associated coherent upper probability function, and m_2 the corresponding Möbius inverse:

$$\underline{P}_2(A) = \max \left\{ 0, 1 - \sum_{x_i \in X_2 \setminus A} u_i' \right\}, \quad \overline{P}_2(A) = 1 - \underline{P}_2(\overline{A}),$$

$$m_2(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \underline{P}_2(B), \quad \forall A \subseteq X_2.$$

We consider $X_0 = \left\{ x_i \in X_2 \mid u_i' = 0 \right\} = \{x_i \in X_2 \mid l_i = u_i\}$.

As shown in the following result, $\underline{P}_1$ is a belief function if, and only if, $\underline{P}_2$ is a belief function.

Theorem 7.2.2 m_1 *is non-negative if, and only if,* m_2 *is non-negative.*

Proof:

For each $A \subseteq X_2 \setminus X_0$, $B \subseteq X_0$, it holds that:

$$\begin{aligned}\underline{P}_1(X_1 \cup B \cup A) &= \max\left\{0, 1 - \sum_{x_i \notin X_1 \cup B \cup A} u'_i\right\} = \max\left\{0, 1 - \sum_{x_i \notin X_1 \cup A} u'_i\right\} \\ &= \underline{P}_1(X_1 \cup A),\end{aligned}$$

since $u'_i = 0 \quad \forall x_i \in B$.

In addition,

$$\begin{aligned}\underline{P}_1(X_1 \cup A) &= \max\left\{0, 1 - \sum_{x_i \notin X_1 \cup A} u'_i\right\} = \max\left\{0, 1 - \sum_{x_i \in X_2 \setminus A} u'_i\right\} \\ &= \underline{P}_2(A), \quad \forall A \subseteq X_2 \setminus X_0.\end{aligned}$$

Consequently, as $\underline{P}_1(C) = 0 \; \forall C$ such that $X_1 \not\subseteq C$ (Corollary 7.2.4), it holds that:

$$\begin{aligned}m_2(A) &= \sum_{A' \subseteq A} (-1)^{|A \setminus A'|} \underline{P}_2(A') \\ &= \sum_{A' \subseteq A} (-1)^{|A \setminus A'|} \underline{P}_1(X_1 \cup A') = m_1(X_1 \cup A), \quad \forall A \subseteq X_2 \setminus X_0.\end{aligned}$$

Let us consider now $B \subseteq X_0$ with $B \neq \emptyset$. For a given $A \subseteq X_2 \setminus X_0$, since $A \cap B = \emptyset$, it is satisfied that $|A \cup B \setminus A' \cup B'| = |A \setminus A'| + |B \setminus B'| \quad \forall A' \subseteq A$, $B' \subseteq B$. Taking into account this issue and that $\underline{P}_1(C) = 0 \; \forall C$ such that $X_1 \not\subseteq C$, we have that:

$$\begin{aligned}m_1(X_1 \cup A \cup B) &= \sum_{A' \subseteq A} \sum_{B' \subseteq B} (-1)^{|(A \cup B) \setminus (A' \cup B')|} \underline{P}_1(X_1 \cup A' \cup B') \\ &= \sum_{A' \subseteq A} (-1)^{|A \setminus A'|} \sum_{B' \subseteq B} (-1)^{|B \setminus B'|} \underline{P}_1(X_1 \cup A' \cup B') \\ &= \sum_{A' \subseteq A} (-1)^{|A \setminus A'|} \sum_{B' \subseteq B} (-1)^{|B \setminus B'|} \underline{P}_1(X_1 \cup A') \\ &= \sum_{A' \subseteq A} (-1)^{|A \setminus A'|} \underline{P}_1(X_1 \cup A') \sum_{j=0}^{|B|} (-1)^j \binom{|B| - j}{j} \\ &= 0, \quad \forall A \subseteq X_2 \setminus X_0,\end{aligned}$$

where we have utilized Lemma 7.2.1 in the last equality taking into account that $|B| \geqslant 1$.

To summarize, $m_1(X_1 \cup A) = m_2(A) \quad \forall A \subseteq X_2 \setminus X_0$ and, if $B \subseteq X_0$ with $|B| \neq \emptyset$, then $m'(X_1 \cup A \cup B) = 0 \quad \forall A \subseteq X_2 \setminus X_0$. Our thesis follows from the previous points. □

We now give a condition that has to be satisfied by m_2 in Case 2 if it is non-negative.

Proposition 7.2.6 *If m_2 is non-negative, then, $\forall A_1, A_2 \subseteq X_2 \setminus X_0$ such that $m_2(A_1) > 0$ and $m_2(A_2) > 0$, it must be satisfied that $|A_1 \setminus A_2| \leqslant 1$.*

Proof: Let $B \subseteq X_2 \setminus X_0$ be a set with smallest cardinality such that $m_2(B) > 0$ (B might not be unique). Then,

$$m_2(B) = \underline{P}_2(B) - \sum_{A \subset B} m_2(A) = \underline{P}_2(B) = 1 - \sum_{x_i \notin B} u_i'$$

$$= 1 - \sum_{x_i \notin B} \sum_{A | x_i \in A} m_2(A) \Rightarrow$$

$$m_2(B) + \sum_{x_i \notin B} \sum_{A | x_i \in A} m_2(A) = 1.$$

Moreover, since $m_2(A) = 0 \quad \forall A \subset B$:

$$1 = \sum_{A \subseteq X_2 \setminus X_0} m_2(A) = m_2(B) + \sum_{A | A \setminus B \neq \emptyset} m_2(A).$$

Thus,

$$1 = m_2(B) + \sum_{x_i \notin B} \sum_{A | x_i \in A} m_2(A) = m_2(B) + \sum_{A | A \setminus B \neq \emptyset} m_2(A)$$

$$\Rightarrow \sum_{A | A \setminus B \neq \emptyset} m_2(A) = \sum_{x_i \notin B} \sum_{A | x_i \in A} m_2(A).$$

In consequence, since $m_2(A) \geqslant 0 \quad \forall A \subseteq X_2 \setminus X_0$, it is not possible that $\exists A \subseteq X_2 \setminus X_0$, $x_i \notin B$, $x_j \notin B$ with $x_i \neq x_j$ such that $m_2(A) > 0$, $x_i \in A$ and $x_j \in A$.

Therefore, if A is a focal element of m_2, i.e, if $m_2(A) > 0$, then there cannot exist x_i, $x_j \in A$ such that $x_i, x_j \notin B \Rightarrow |A \setminus B| \leqslant 1 \quad \forall A \subseteq X_2 \setminus X_0$ such that $m_2(A) > 0$. As B is a focal element with minimum cardinality, it is immediate to conclude that $|A_1 \setminus A_2| \leqslant 1 \quad \forall A_1, A_2 \subseteq X_2 \setminus X_0$ such that $m_2(A_1) > 0$ and $m_2(A_2) > 0$. □

Hence, if the Möbius inverse associated with the natural extension of $\mathcal{I}_2$ is greater or equal than 0, then the difference between two focal elements cannot have a cardinality greater than 1. In the following result, we give a necessary condition that the probability intervals of $\mathcal{I}_2$ have to satisfy for the Möbius inverse to be non-negative: the sum of each pair of upper extremes of the intervals must be greater than 1.

Proposition 7.2.7 *If* m_2 *is non-negative, then*

$$u'_i + u'_j > 1, \quad \forall x_i, x_j \in X_2 \setminus X_0.$$

Proof: Let us assume that m_2 is non-negative. Let $H \subseteq X_2 \setminus X_0$ be a set of maximum cardinality such that $\sum_{x_o \in H} u'_o \leqslant 1$. Suppose that our thesis does not hold, which is equivalent to $|H| \geqslant 2$.

Since $\sum_{x_j \in X_2 \setminus X_0} u'_j > 1$, $\exists x_k \notin H$, $x_k \in X_2 \setminus X_0$. Now, due to Proposition 2.2.7,

$$\overline{P}_2(H \cup \{x_k\}) = \min \left\{ \sum_{x_o \in H \cup \{x_k\}} u'_o, 1 \right\} = 1,$$

because H is a set of maximum cardinality that satisfies $\sum_{x_o \in H} u'_o \leqslant 1$.

Clearly, $\overline{P}_2(\{x_k\}) = u'_k, \quad \overline{P}_2(H) = \min\left\{\sum_{x_o \in H} u'_o, 1\right\} = \sum_{x_o \in H} u'_o$.

In addition, $u'_k + \sum_{x_o \in H} u'_o > 1$. Thus,

$$\overline{P}_2(H \cup \{x_k\}) < \overline{P}_2(H) + \overline{P}_2(\{x_k\}) \Rightarrow$$

$$\sum_{A | A \cap (H \cup \{x_k\}) \neq \emptyset} m_2(A) < \sum_{A | A \cap H \neq \emptyset} m_2(A) + \sum_{A | x_k \in A} m_2(A),$$

which implies that $\exists\ C_{ik} \subseteq X_2 \setminus X_0$ such that $m_2(C_{ik}) > 0$, $x_k \in C_{ik}$, and $x_i \in C_{ik}$, for some $x_i \in H$.

Since $|H| \geqslant 2$, $\exists x_j \in H$ such that $x_j \neq x_i$. There exists a subset $C_j \subseteq X_2 \setminus X_0$ satisfying $x_j \in C_j$ and $m_2(C_j) > 0$ because $u'_j > 0$.

Furthermore, $m_2(\overline{H}) = \underline{P}_2(\overline{H}) = 1 - \sum_{x_o \in H} u'_o$ because H is a subset of maximum cardinality such that $\sum_{x_o \in H} u'_o \leqslant 1$. Two cases are distinguished:

1. If $\sum_{x_o \in H} u'_o = 1$, then:

$$\sum_{x_o \in H} \sum_{A | x_o \in A} m_2(A) = \sum_{A \subseteq X_2 \setminus X_0} m_2(A).$$

Consequently, it is not possible that $\exists C \subseteq X_2 \setminus X_0$ such that $m_2(C) > 0$ and $x_o, x_u \in C$ for some $x_o, x_u \in H$.

2. If $\sum_{x_o \in H} u'_o < 1$, then:

$$m_2(\overline{H}) = 1 - \sum_{x_o \in H} u'_o > 0,$$

which implies that there can not exist a focal element C with $x_o, x_u \in C \cap H$ as, in that case, $x_o, x_u \in C \setminus \overline{H}$, which contradicts Proposition 7.2.6.

Therefore, a focal element can never contain two elements belonging to H. In consequence, $x_i \notin C_j$. It implies that $x_k \in C_j$ since, otherwise, $\{x_i, x_k\} \subseteq C_{ik} \setminus C_j$.

Given a focal set C, we distinguish two cases:

1. If $x_i \notin C$, then $x_k \in C$, because, else, $\{x_i, x_k\} \subseteq C_{ik} \setminus C$.
2. If $x_i \in C$, then $x_j \notin C$. As $x_j \in C_j$ and $x_k \in C_j$, it holds that $x_k \in C$ since, otherwise, $\{x_k, x_j\} \subseteq C_j \setminus C$.

Hence, we have that $x_k \in C \quad \forall C \subseteq X_2 \setminus X_0$ such that $m_2(C) > 0$, which implies that $u'_k = 1$. But $u'_o < 1 \quad \forall x_o \in X_2 \setminus X_0$, and we get a contradiction. Thereby, if m_2 is non-negative, then $|H| \leqslant 1 \Rightarrow u'_i + u'_j > 1, \quad \forall x_i, x_j \in X_2 \setminus X_0$. □

If the condition of Proposition 7.2.7 is satisfied, then all the subsets of $X_2 \setminus X_0$ whose cardinality is lower or equal than $|X_2 \setminus X_0| - 2$ have a Möbius inverse equal to 0. The Möbius inverse for sets of cardinality equal to $|X_2 \setminus X_0| - 1$ is non-negative (Corollary 7.2.2). Thus, the natural extension of $\mathcal{I}_2$ is a belief function if, and only if, the Möbius inverse for $X_2 \setminus X_0$ is greater or equal than 0. In the following proposition, we show the condition that the intervals of $\mathcal{I}_2$ must verify, assuming that they satisfy the property of Proposition 7.2.7, for the corresponding Möbius inverse to be non-negative.

Proposition 7.2.8 *If it is satisfied that*

$$u'_i + u'_j > 1 \quad \forall x_i, x_j \in X_2 \setminus X_0,$$

then m_2 *in non-negative if, and only if,*

$$\sum_{x_i \in X_2 \setminus X_0} u'_i \geqslant |X_2 \setminus X_0| - 1.$$

Proof:

Under our hypothesis,

$$m_2\,(A) = 0 \quad \forall A \subseteq X_2 \setminus X_0, \quad |A| \leqslant |X_2 \setminus X_0| - 2.$$

For sets of cardinality equal to $|X_2 \setminus X_0| - 1$:

$$m_2\,(X_2 \setminus \{X_0 \cup \{x_i\}\}) = 1 - u'_i > 0, \quad \forall x_i \in X_2 \setminus X_0.$$

Hence, $m_2(A) \geqslant 0 \; \forall A \subset X_2 \setminus X_0$. Now,

$$m_2\,(X_2 \setminus X_0) = 1 - \sum_{A \subset X_2 \subset X_0} m_2(A) = 1 - \sum_{x_i \in X_2 \setminus X_0} m_2\,(X_2 \setminus \{X_0 \cup \{x_i\}\})$$
$$= 1 - \sum_{x_i \in X_2 \setminus X_0} (1 - u'_i) = 1 - |X_2 \setminus X_0| + \sum_{x_i \in X_2 \setminus X_0} u'_i.$$

Therefore, m_2 is non-negative if, and only if,

$$m_2\left(X_2 \setminus X_0\right) \geqslant 0 \Leftrightarrow 1 - |X_2 \setminus X_0| + \sum_{x_i \in X_2 \setminus X_0} u_i' \geqslant 0$$

$$\Leftrightarrow \sum_{x_i \in X_2 \setminus X_0} u_i' \geqslant |X_2 \setminus X_0| - 1.$$

□

The following result summarizes the necessary and sufficient conditions that a given reachable set of probability intervals has to satisfy to be representable by a belief function.

Theorem 7.2.3 *Let* $X = \{x_1, x_2, \ldots, x_t\}$ *be a finite set and* $\mathcal{I} = \{[l_i, u_i], \quad i = 1, 2, \ldots, t\}$ *a reachable set of probability intervals on* X. *Let us consider:*

$$L = \sum_{i=1}^{t} l_i, \quad X_0 = \{x_i \in X \mid l_i = u_i,\ 1 \leqslant i \leqslant t\},$$

$$X_1 = \left\{ x_i \in X \mid u_i + \sum_{j=1, j \neq i}^{t} l_j = 1,\ 1 \leqslant i \leqslant t \right\},$$

$$X_2 = \left\{ x_i \in X \mid u_i + \sum_{j=1, j \neq i}^{t} l_j < 1,\ 1 \leqslant i \leqslant t \right\}.$$

Then, $\mathcal{I}$ *can be represented by a belief function if, and only if, one of the two following conditions is satisfied:*

1. $\sum_{x_i \in X_1} l_i \leqslant 1 - \sum_{x_i \in X_2} u_i$.
2. $\sum_{x_i \in X_1} l_i > 1 - \sum_{x_i \in X_2} u_i$, *and the following two statements hold:*
   a) $u_i + u_j > 1 - \sum_{k=1, k \neq i, k \neq j}^{t} l_k, \quad \forall x_i, x_j \in X_2 \setminus X_0,$
   b) $\sum_{x_i \in X_2 \setminus X_0} u_i \geqslant (1 - L)\left(|X_2 \setminus X_0| - 1\right) + \sum_{x_i \in X_2 \setminus X_0} l_i$.

Indeed, this theorem does not explicitly assume that $L < 1$, as the previous results. Nevertheless, it is easy to observe that, when $L = 1$, $X_1 = X_0 = X$, and $X_2 = \emptyset$. Hence, if $L = 1$, then $\sum_{x_i \in X_1} l_i = 1 - \sum_{x_i \in X_2} u_i$ and, consequently, $\mathcal{I}$ can be represented by a belief function. Moreover, it holds that $X_0 \subseteq X_2$ whenever $L < 1$.

In this way, given a reachable set of probability intervals on a finite set, we can divide the set into two subsets. The first one is composed of those elements whose upper probability is equal to one minus the sum of the lower probabilities of the remaining elements. The second subset is the complementary set of the first one. We also consider a subset composed of those elements whose lower and upper probabilities coincide. Then, Theorem 7.2.3 gives the necessary and sufficient conditions that the sums of the lower and upper probability values on these subsets have to satisfy for the Möbius inverse associated with the natural extension of the set of intervals to be non-negative.

7.2.1.1 *An example: The Imprecise Dirichlet Model*

Let X be a discrete variable whose set of possible values is $\{x_1, x_2, \dots, x_t\}$. Suppose that we have a sample of N independent and identically distributed observations about X. For each $i = 1, 2, \dots, t$, let $n(x_i)$ denote the number of observations of the x_i value in the sample. As shown in Section 2.3.1, we have the following set of IDM probability intervals on X:

$$\mathcal{I}_{IDM} = \left\{ \left[l_i^{IDM}, u_i^{IDM} \right], \quad i = 1, 2, \dots, t \right\},$$

where $l_i^{IDM} = \frac{n(x_i)}{N+s}$ and $u_i^{IDM} = \frac{n(x_i)+s}{N+s}$, $\forall i = 1, 2, \dots, t$, s being the IDM parameter.

As pointed out in Section 2.3.1, $\mathcal{I}_{IDM}$ is reachable and can be represented by a belief function.

Indeed, using Theorem 3, we have that $X_1 = X$ and $X_0 = X_2 = \emptyset$ since:

$$\begin{aligned} u_i^{IDM} + \sum_{j=1, j\neq i}^{t} l_j^{IDM} &= \frac{n(x_i)+s}{N+s} + \sum_{j=1, j\neq i}^{t} \frac{n(x_j)}{N+s} \\ &= \frac{N+s}{N+s} = 1, \quad \forall i = 1, 2, \dots, t. \end{aligned}$$

Consequently,

$$\sum_{x_i \in X_1} l_i^{IDM} = \sum_{i=1}^{t} l_i^{IDM} = \sum_{i=1}^{t} \frac{n(x_i)}{N+s} = \frac{N}{N+s} < 1 = 1 - \sum_{x_i \in X_2} u_i.$$

7.2.2 Belief functions representable via reachable probability intervals

Let Bel be a belief function on X and Pl its associated plausibility function. Let m denote the BPA corresponding to Bel, computed through Equation (2.27). Let us consider the set of belief intervals for singletons:

$$\mathcal{I}_{Bel} = \{ [Bel(\{x_i\}), Pl(\{x_i\})], \quad i = 1, 2, \dots, t \}. \tag{7.4}$$

Wang and Song [212] demonstrated that $\mathcal{I}_{Bel}$ is always reachable. The credal set composed of all probability distributions consistent with $\mathcal{I}_{Bel}$ is given by:

$$\mathcal{P}(\mathcal{I}_{Bel}) = \{p \in \mathcal{P}(X) \mid Bel(\{x_i\}) \leqslant p(\{x_i\}) \leqslant Pl(\{x_i\}), \quad \forall i = 1, 2, \ldots, t\}, \tag{7.5}$$

$\mathcal{P}(X)$ being the set of all probability distributions on X.

Let $\mathcal{P}_{Bel}$ denote the credal set associated with Bel:

$$\mathcal{P}_{Bel} = \{p \in \mathcal{P}(X) \mid Bel(A) \leqslant p(A) \quad \forall A \subseteq X\}. \tag{7.6}$$

In this way, the belief function Bel can be represented by a reachable set of probability intervals if, and only if, $\mathcal{P}(\mathcal{I}_{Bel}) = \mathcal{P}_{Bel}$. It is easy to observe that it always holds that $\mathcal{P}_{Bel} \subseteq \mathcal{P}(\mathcal{I}_{Bel})$. However, it is not always satisfied that $\mathcal{P}(\mathcal{I}_{Bel}) \subseteq \mathcal{P}_{Bel}$, as can be observed in Example 2.2.1.

The following result shows that the belief function Bel is representable by a reachable set of probability intervals if, and only if, it coincides with the natural extension of $\mathcal{I}_{Bel}$.

Theorem 7.2.4 $\mathcal{P}(\mathcal{I}_{Bel}) = \mathcal{P}_{Bel} \Leftrightarrow$
$Bel(A) = \max\left\{\sum_{x_i \in A} Bel(\{x_i\}), 1 - \sum_{x_i \notin A} Pl(\{x_i\})\right\}, \quad \forall A \subseteq X.$

Proof: It immediately follows from Proposition 2.2.7. □

We must remark that the condition given in Theorem 7.2.4 might be computationally hard to check as it requires checking the equality for each $A \subseteq X$, and the number of subsets exponentially grows as the number of alternatives increases.

For this reason, in this section, we aim to provide a necessary and sufficient condition for Bel to be representable via belief intervals for singletons in terms of the relations between the focal elements of m.

We consider the following BPA on X, m′:

$$\begin{aligned} m'(\{x_i\}) &= 0, \quad \forall i = 1, 2, \ldots, t, \\ m'(A) &= \frac{m(A)}{1 - M}, \quad \forall A \subseteq X, |A| \geqslant 2, \end{aligned} \tag{7.7}$$

where $M = \sum_{i=1}^{t} m(\{x_i\})^2$.

The following proposition shows that m′ is well-defined as a BPA on X.

Proposition 7.2.9 *The set-function m′, defined in Equation (7.7), is well-defined as a BPA on X.*

[2] Here, we do not consider the case $M = 1$ because, in that situation, m is a probability distribution.

Proof: As m is a BPA on X, it holds that:

$$(m(A) \geqslant 0) \wedge (1 - M \geqslant 0) \Rightarrow m'(A) \geqslant 0 \quad \forall A \subseteq X, |A| \geqslant 2,$$

$$\begin{aligned} & m(A) + M \leqslant \sum_{B \subseteq X} m(B) = 1 \Rightarrow m(A) \leqslant 1 - M \Rightarrow \\ & m'(A) \leqslant 1 \quad \forall A \subseteq X,\ |A| \geqslant 2, \end{aligned}$$

$$\sum_{A \subseteq X} m'(A) = \sum_{A \subseteq X, |A| \geqslant 2} \frac{m(A)}{1 - M} = \frac{1 - \sum_{i=1}^{t} m\left(\{x_i\}\right)}{1 - M} = 1.$$

□

All focal elements of m' have a cardinality greater or equal than 2. Among the non-singleton subsets, the focal elements of m' coincide with the ones of m. It is expressed in the following result, whose proof is immediate.

Proposition 7.2.10 $\forall A \subseteq X$ *such that* $|A| \geqslant 2$, $m(A) > 0 \Leftrightarrow m'(A) > 0$.

Let $\mathrm{Bel}_{m'}$ and $\mathrm{Pl}_{m'}$ denote, respectively, the belief and plausibility functions associated with m'. Let $\mathcal{P}\left(\mathrm{Bel}_{m'}\right)$ denote the credal set corresponding to $\mathrm{Bel}_{m'}$ and $\mathcal{P}\left(\mathcal{I}_{\mathrm{Bel}_{m'}}\right)$ the credal set consistent with the belief intervals for singletons associated with m'. The following proposition shows that Bel represents the same uncertainty-based information as its associated belief intervals for singletons if, and only if, the same occurs with $\mathrm{Bel}_{m'}$.

Proposition 7.2.11 $\mathcal{P}\left(\mathrm{Bel}\right) = \mathcal{P}(\mathcal{I}_{\mathrm{Bel}}) \Leftrightarrow \mathcal{P}\left(\mathrm{Bel}_{m'}\right) = \mathcal{P}\left(\mathcal{I}_{\mathrm{Bel}_{m'}}\right)$.

Proof: For each $A \subseteq X$, it is satisfied that:

$$\mathrm{Bel}_{m'}(A) = \max\left\{ \sum_{x_i \in A} \mathrm{Bel}_{m'}\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} \mathrm{Pl}_{m'}\left(\{x_i\}\right) \right\} \Leftrightarrow$$

$$\sum_{B \subseteq A} m'(B) = \max\left\{ \sum_{x_i \in A} m'\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} \sum_{B | x_i \in B} m'(B) \right\} \Leftrightarrow$$

$$\sum_{B \subseteq A, |B| \geqslant 2} \frac{m(B)}{1-M} = \max\left\{ 0, 1 - \sum_{x_i \notin A} \sum_{B | x_i \in B \wedge |B| \geqslant 2} \frac{m(B)}{1-M} \right\} \Leftrightarrow$$

$$\sum_{B \subseteq A, |B| \geqslant 2} m(B) = \max\left\{ 0, 1 - M - \sum_{x_i \notin A} \sum_{B | x_i \in B \wedge |B| \geqslant 2} m(B) \right\} \Leftrightarrow$$

$$\sum_{x_i \in A} m\left(\{x_i\}\right) + \sum_{B \subseteq A, |B| \geqslant 2} m(B) = \max\left\{ \sum_{x_i \in A} m\left(\{x_i\}\right), \right.$$

$$\left. 1 - M + \sum_{x_i \in A} m\left(\{x_i\}\right) - \sum_{x_i \notin A} \sum_{B | x_i \in B \wedge |B| \geqslant 2} m(B) \right\} \Leftrightarrow$$

$$\sum_{B \subseteq A} m(B) = \max\left\{ \sum_{x_i \in A} \mathrm{Bel}\left(\{x_i\}\right), 1 - \right.$$

$$\left. \sum_{x_i \notin A} \left[m\left(\{x_i\}\right) + \sum_{B | x_i \in B \wedge |B| \geqslant 2} m(B) \right] \right\} \Leftrightarrow$$

$$\mathrm{Bel}(A) = \max\left\{ \sum_{x_i \in A} \mathrm{Bel}\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} \sum_{B | x_i \in B} m(B) \right\} \Leftrightarrow$$

$$\mathrm{Bel}(A) = \max\left\{ \sum_{x_i \in A} \mathrm{Bel}\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} \mathrm{Pl}\left(\{x_i\}\right) \right\},$$

and our thesis follows from Theorem 7.2.4.

□

Hence, we focus on studying when $\mathrm{Bel}_{m'}$ can be represented via its corresponding set of belief intervals for singletons.

The following theorem gives the necessary and sufficient condition for $\mathrm{Bel}_{m'}$ to represent the same uncertainty-based information as its corresponding set of belief intervals for singletons: the difference between each pair of focal elements of m' has a cardinality lower than 2.

Theorem 7.2.5 *$\mathcal{P}\left(\mathcal{I}_{Bel_{m'}}\right) = \mathcal{P}(Bel_{m'}) \Leftrightarrow |B_1 \setminus B_2| \leqslant 1 \ \forall B_1, B_2 \subseteq X$ such that $m'(B_1) > 0$ and $m'(B_2) > 0$.*

Proof: Suppose that $\mathcal{P}\left(\mathcal{I}_{Bel_{m'}}\right) = \mathcal{P}(Bel_{m'})$. Let A be a focal element of m'. From Theorem 7.2.4, it follows that:

$$\begin{aligned} Bel_{m'}(A) &= \max\left\{\sum_{x_i \in A} Bel_{m'}(\{x_i\}), 1 - \sum_{x_i \notin A} Pl_{m'}(\{x_i\})\right\} \\ &= \max\left\{\sum_{x_i \in A} m'(\{x_i\}), 1 - \sum_{x_i \notin A} \sum_{B | x_i \in B} m'(B)\right\} \\ &= \max\left\{0, 1 - \sum_{B \subseteq X} m'(B)\, |B \setminus A|\right\} = 1 - \sum_{B \subseteq X} m'(B)\, |B \setminus A|. \end{aligned}$$

The last equality is because A is a focal element of m'.

In consequence, $\sum_{B \subseteq A} m'(B) = 1 - \sum_{B \subseteq X} m'(B)\, |B \setminus A|$.

Furthermore,

$$1 = \sum_{B \subseteq X} m'(B) = \sum_{B \subseteq A} m'(B) + \sum_{B | B \setminus A \neq \emptyset} m'(B).$$

Therefore,

$$\begin{aligned} \sum_{B \subseteq A} m'(B) &= 1 - \sum_{B \subseteq X} m'(B)\, |B \setminus A| \\ &= \sum_{B \subseteq A} m'(B) + \sum_{B | B \setminus A \neq \emptyset} m'(B) - \sum_{B \subseteq X} m'(B)\, |B \setminus A|, \end{aligned}$$

which implies that

$$\sum_{B | B \setminus A \neq \emptyset} m'(B) = \sum_{B \subseteq X} m'(B)\, |B \setminus A|.$$

Thus, if $m'(B) > 0$, then it is not possible that $|B \setminus A| \geqslant 2$. It can be concluded that $|B_1 \setminus B_2| \leqslant 1 \quad \forall B_1, B_2 \subseteq X$ such that $m'(B_1) > 0$ and $m'(B_2) > 0$.

Let us assume now that $|B_1 \setminus B_2| \leqslant 1 \quad \forall B_1, B_2 \subseteq X$ such that $m'(B_1) > 0$ and $m'(B_2) > 0$. Let us consider $A \subseteq X$. We have that:

$$\begin{aligned}
&\max\left\{\sum_{x_i \in A} Bel_{m'}(\{x_i\}), 1 - \sum_{x_i \notin A} Pl_{m'}(\{x_i\})\right\} = \\
&\max\left\{\sum_{x_i \in A} m'(\{x_i\}), 1 - \sum_{x_i \notin A} \sum_{B | x_i \in B} m'(B)\right\} = \\
&\max\left\{0, 1 - \sum_{B \subseteq X} m'(B)\,|B \setminus A|\right\} = \\
&\max\left\{0, \sum_{B \subseteq A} m'(B) + \sum_{B | B \setminus A \neq \emptyset} m'(B) - \sum_{B \subseteq X} m'(B)\,|B \setminus A|\right\}.
\end{aligned}$$

We distinguish two cases:

- **Case 1:** $\exists C \subseteq A$ **such that** $m'(C) > 0$.

  By hypothesis, it holds that, if $B \subseteq X$ satisfies that $|B \setminus C| > 1$, then $m'(B) = 0$. Hence, since $C \subseteq A$, $m'(B) = 0 \quad \forall B \subseteq X$ such that $|B \setminus A| > 1$. Consequently, $|B \setminus A| \leqslant 1 \ \forall B \subseteq X$ such that $m'(B) > 0$. Then,

$$\begin{aligned}
&\max\left\{\sum_{x_i \in A} Bel_{m'}(\{x_i\}), 1 - \sum_{x_i \notin A} Pl_{m'}(\{x_i\})\right\} = \\
&\max\left\{0, \sum_{B \subseteq A} m'(B) + \sum_{B | B \setminus A \neq \emptyset} m'(B) - \sum_{B \subseteq X} m'(B)\,|B \setminus A|\right\} = \\
&\max\left\{0, \sum_{B \subseteq A} m'(B)\right\} = \sum_{B \subseteq A} m'(B) = Bel_{m'}(A).
\end{aligned}$$

- **Case 2:** $m'(B) = 0 \quad \forall B \subseteq A$.

In this case, $Bel_{m'}(A) = 0$ and

$$\max\left\{\sum_{x_i \in A} Bel_{m'}\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} Pl_{m'}\left(\{x_i\}\right)\right\} =$$

$$\max\left\{0, \sum_{B \subseteq A} m'(B) + \sum_{B | B \setminus A \neq \emptyset} m'(B) - \sum_{B \subseteq X} m'(B)\,|B \setminus A|\right\} =$$

$$\max\left\{0, \sum_{B | B \setminus A \neq \emptyset} m'(B)\,(1 - |B \setminus A|)\right\} = 0 = Bel_{m'}(A).$$

Thereby,

$$\max\left\{\sum_{x_i \in A} Bel_{m'}\left(\{x_i\}\right), 1 - \sum_{x_i \notin A} Pl_{m'}\left(\{x_i\}\right)\right\} = Bel_{m'}(A), \quad \forall A \subseteq X,$$

and, from Theorem 7.2.4, we conclude that $\mathcal{P}\left(\mathcal{I}_{Bel_{m'}}\right) = \mathcal{P}\left(Bel_{m'}\right)$.

□

As a consequence of this theorem and Proposition 7.2.10, the necessary and sufficient condition for a given belief function on X to be representable by a reachable set of probability intervals is expressed in the following corollary:

Corollary 7.2.7 *Let* Bel *be a belief function on a finite set* X *and* m *its associated BPA. Let* $\mathcal{P}\left(Bel\right)$ *denote the credal set associated with* Bel *and* $\mathcal{P}(\mathcal{I}_{Bel})$ *the credal set compatible with the belief intervals for singletons derived from* Bel*. It holds that*

$\mathcal{P}\left(Bel\right) = \mathcal{P}(\mathcal{I}_{Bel}) \Leftrightarrow |B_1 \setminus B_2| \leqslant 1 \quad \forall B_1, B_2 \subseteq X$ *such that* $m(B_i) > 0$ *and* $|B_i| \geqslant 2$*, for* $i = 1, 2$.

Hence, the belief function associated with the BPA m given in Example 2.2.1 cannot be represented via a reachable set of probability intervals because $\{x_1, x_2\}$ and $\{x_3, x_4\}$ are focal elements of m and $|\{x_3, x_4\} \setminus \{x_1, x_2\}| = 2$.

We show below another example where the belief function represents the same uncertainty-based information as its corresponding set of belief intervals for singletons.

Example 7.2.3 *Let* $X = \{x_1, x_2, x_3, x_4\}$ *be a finite set and* m *the following BPA on* X*:*

$$m\left(\{x_3\}\right) = 0.3, \quad m\left(\{x_1, x_2\}\right) = 0.3,$$
$$m\left(\{x_1, x_2, x_3\}\right) = 0.1, \quad m\left(\{x_1, x_2, x_4\}\right) = 0.3.$$

The non-singleton focal elements of m *are* $\{x_1, x_2\}$, $\{x_1, x_2, x_3\}$, *and* $\{x_1, x_2, x_4\}$. *We can check that* $|\{x_1, x_2\} \setminus \{x_1, x_2, x_i\}| = 0$, $|\{x_1, x_2, x_i\} \setminus \{x_1, x_2\}| = 1$, *and* $\left|\{x_1, x_2, x_i\} \setminus \left\{x_1, x_2, x_j\right\}\right| = 1$, *for* $i, j \in \{3, 4\}$.

Thus, in this case, the set of probability distributions compatible with the belief function corresponding to m *coincides with the set of probability distributions consistent with the belief intervals for singletons.*

If there is a unique non-singleton focal element of m, namely B, then $m'(B) = 1$ and, clearly, Bel y $\mathrm{Bel}_{m'}$ can be represented via reachable sets of probability intervals. Also, if all the focal elements of m are singletons, then m is a probability distribution.

For testing the condition given in Corollary 7.2.7, it is just necessary to check whether there exists, among the non-singletons subsets, a focal element of greatest cardinality such that its difference with another focal element of smallest cardinality has a cardinality greater than one. Consequently, that condition might be easier to check than the one given in Theorem 7.2.4.

From Corollary 7.2.7, it is easy to deduce that, if there are three or fewer alternatives, then Bel always represents the same uncertainty-based information as its associated set of belief intervals for singletons. Therefore, we have the following result:

Corollary 7.2.8 *Let* Bel *be a belief function on a finite set* $X = \{x_1, \ldots, x_t\}$ *with* $t \leqslant 3$. *Let* $\mathcal{P}(\mathrm{Bel})$ *denote the credal set correspoding to* Bel *and* $\mathcal{P}(\mathcal{I}_{\mathrm{Bel}})$ *the credal set associated with the corresponding set of belief intervals for singletons. Then, it is always satisfied that* $\mathcal{P}(\mathcal{I}_{\mathrm{Bel}}) = \mathcal{P}(\mathrm{Bel})$.

7.2.2.1 *Particular cases of belief functions representable by reachable sets of probability intervals*

Within this subsection, we use the characterization given in Corollary 7.2.7 to describe some special types of belief functions representable by means of reachable probability intervals.

p-boxes: Suppose that we have a finite set $X = \{x_1, x_2, \ldots, x_t\}$ that is also ordered, i.e, $x_1 < x_2 < \ldots < x_t$. Let $(\underline{F}, \overline{F})$ be a p-box on X. Let Bel_F be the belief function derived from $(\underline{F}, \overline{F})$, computed through Equation (2.44), and m_{Bel_F} the BPA corresponding to Bel_F, determined via Equation (2.27).

As pointed out in Section 2.2.4.5, the focal elements of the BPA associated with a p-box are always intervals ordered, following the interval order given by Equation (2.45).

The result presented below demonstrates that, for a p-box to be representable via reachable probability intervals, it is not possible that there are more than three non-singleton focal elements of the corresponding BPA:

Proposition 7.2.12 *If $A_1, A_2, \dots, A_k$, with $k \geqslant 4$, are the non-singleton focal elements of Bel_F, then $\left(\underline{F}, \overline{F}\right)$ is not representable via a rechable set of probability intervals.*

Proof: Since Bel_F is the belief function associated with a p-box, its non-singleton focal elements are intervals ordered, that is, $A_1 \preceq A_2 \preceq \dots \preceq A_k$.

Let us denote $A_i = [a_i, b_i]$, with $a_i, b_i \in X, \quad \forall i = 1, 2, \dots, k$.

Three cases are distinguished:

1. If $a_4 > a_1 + 1$, then $|A_1 \setminus A_4| \geqslant 2$. Likewise, $|A_4 \setminus A_1| \geqslant 2$ whenever $b_4 > b_1 + 1$.
2. If $a_4 = a_1$, then $a_1 = a_2 = a_3 = a_4$. Clearly, in this case, $A_1 \subset A_2 \subset A_3 \subset A_4$, and $|A_3 \setminus A_1| \geqslant 2$.
3. Analogously, if $b_1 = b_4$ then, $b_1 = b_2 = b_3 = b_4$ and, obviously, $A_4 \subset A_3 \subset A_2 \subset A_1$. In consequence, $|A_2 \setminus A_4| \geqslant 2$.

Let us prove that, under our hypothesis, these are the only possible situations. Indeed, if $a_4 = a_1 + 1$ and $b_4 = b_1 + 1$, then there are two possibilities:

1. If $a_1 = a_2$, then $b_2 = b_1 + 1 = b_3 = b_4$. In this case, it is not possible that $A_2 \neq A_3$ and $A_3 \neq A_4$, and we get a contradiction.
2. If $a_2 = a_1 + 1$, then $a_2 = a_3 = a_4$. This contradicts that $A_2 \neq A_3$ and $A_3 \neq A_4$.

Therefore, under our hypothesis, there always exists A_i, A_j such that $\left|A_i \setminus A_j\right| \geqslant 2$, with $\{i, j\} \subseteq \{1, 2, 3, 4\}$. Corollary 7.2.7 allows us to conclude that Bel_F does not represent the same uncertainty-based information as its associated set of belief intervals for singletons.

□

Hence, for a p-box to be representable via a reachable set of probability intervals, it cannot have more than three non-singleton focal elements. The following result shows the necessary and sufficient condition for a p-box with three non-singleton focal elements to represent the same uncertainty-based information as its corresponding set of belief intervals for singletons:

Proposition 7.2.13 *Suppose that the belief function* Bel_F *has three non-singleton focal elements* $A_1 \preceq A_2 \preceq A_3$. *Let us denote* $A_i = [a_i, b_i]$, *for* $i = 1, 2, 3$. Bel_F *is representable by a rechable set of probability intervals if, and only if,* $a_3 = a_1 + 1$ *and* $b_3 = b_1 + 1$.

Proof:

- If $a_3 > a_1 + 1$, then $|A_1 \setminus A_3| \geqslant 2$. Analogously, $|A_3 \setminus A_1| \geqslant 2$ whenever $b_3 > b_1 + 1$.
- If $a_3 = a_1$ then, $A_1 \subset A_2 \subset A_3$, which implies that $|A_3 \setminus A_1| \geqslant 2$. Likewise, if $b_3 = b_1$, then $b_1 = b_2 = b_3$. In this case, $A_3 \subset A_2 \subset A_1$ and $|A_1 \setminus A_3| \geqslant 2$.

Thereby, due to Corollary 7.2.7, for Bel_F to be representable via a rechable set of probability intervals, it is necessary that $a_3 = a_1 + 1$ and $b_3 = b_1 + 1$. Let us show that such a condition is sufficient.

In that situation, $A_1 \setminus A_3 = \{a_1\}, \quad A_3 \setminus A_1 = \{b_3\}$, and two cases are distinguished:

1. If $a_1 = a_2$, then $b_2 = b_1 + 1 = b_3$. It holds that $A_1 \subset A_2$ and $A_2 \setminus A_1 = \{b_2\}$. Also, $a_3 = a_2 + 1 \Rightarrow A_3 \subset A_2$ and $A_2 \setminus A_3 = \{a_2\}$.
2. If $a_2 = a_1 + 1$, then $a_2 = a_3$ and $b_3 = b_2 + 1 \Rightarrow b_1 = b_2$, which implies that $A_2 \subset A_1$ with $A_1 \setminus A_2 = \{a_1\}$ and $A_2 \subset A_3$ with $A_3 \setminus A_2 = \{b_3\}$.

□

In this way, the characterization given by Corollary 7.2.7 lets us easily check when the belief function corresponding to a p-box is representable via a reachable set of probability intervals. Indeed, we only need to check when there are three or fewer non-singleton focal elements.

Possibility measures: Let $X = \{x_1, x_2, \ldots, x_t\}$ be a finite set, Poss a possibility measure on X and Necc its associated necessity measure. Let m_{Necc} denote the BPA corresponding to Necc, computed through Equation (2.27).

As commented in Section 2.2.4.6, the focal elements of m_{Necc} must be nested. Therefore, it is very easy to deduce that, for a possibility measure to be representable through a reachable set of probability intervals, it cannot have more than two non-singleton focal elements. When we have only two non-singleton focal elements, it is sufficient to check whether there are two or more elements that belong to one of such sets but not to the other one.

The Imprecise Dirichlet Model Let X be a discrete variable whose possible values are $\{x_1, x_2, \ldots, x_t\}$. Suppose that there is sample of N independent and identically distributed observations about X. Let $n(x_i)$ denote the number of observations of the x_i value in the sample, $\forall i = 1, 2, \ldots, t$. As shown in Section 2.3.1, the IDM can also be determined via the following BPA on $\{x_1, x_2, \ldots, x_t\}$:

$$\begin{aligned}
&m^{IDM}(\{x_i\}) = \frac{n(x_i)}{N+s}, \quad \forall i = 1, 2, \ldots, t, \\
&m^{IDM}(A) = 0, \quad \forall A \subseteq \{x_1, x_2, \ldots, x_t\} \mid 2 \leqslant |A| < t, \\
&m^{IDM}(\{x_1, x_2, \ldots, x_t\}) = \frac{s}{N+s},
\end{aligned}$$

where s is the IDM parameter.

We may deduce that the only non-singleton focal element of m^{IDM} is the total set. In consequence, according to Corollary 7.2.7, the belief function associated with m^{IDM} can be represented by means of a reachable set of probability intervals. Actually, such a set coincides with the set of IDM probability intervals, defined in Equation (2.59).

7.3 Properties of A-NPI-M credal sets

Let X be a discrete variable and $\{x_1, x_2, \ldots, x_t\}$ its set of possible values. Suppose that we have a sample of N independent and identically distributed outcomes of X. Let $n(x_i)$ denote the number of occurrences of x_i in the sample, $\forall i = 1, 2, \ldots, t$. Let t_{obs} (t_{unobs}) be the number of observed (unobserved) values of X in the sample:

$$\begin{aligned}
t_{obs} &= |\{x_i : n(x_i) > 0, \quad i = 1, 2, \ldots, t\}|, \\
t_{unobs} &= |\{x_i : n(x_i) = 0, \quad i = 1, 2, \ldots, t\}|.
\end{aligned}$$

As shown in Section 2.3.4, we have the following set of A-NPI-M probability intervals on X (See Equation (2.68)):

$$\mathcal{I}_{ANPI} = \left\{ \left[\max\left(\frac{n(x_i) - 1}{N}, 0 \right), \min\left(\frac{n(x_i) + 1}{N}, 1 \right) \right], \quad i = 1, 2, \ldots, t \right\}.$$

The set of A-NPI-M probability intervals gives rise to the following credal set on X, determined via Equation (2.69):

$$\begin{aligned}
&\mathcal{P}(\mathcal{I}_{ANPI}) = \{p \in \mathcal{P}(X) \mid p(x_i) \in \\
&\left[\max\left(\frac{n(x_i) - 1}{N}, 0 \right), \min\left(\frac{n(x_i) + 1}{N}, 1 \right) \right], \quad \forall i = 1, 2, \ldots, t \right\}.
\end{aligned}$$

For each $A \subseteq \{x_1, x_2, \dots, x_t\}$, let $n(A)$ denote the number of observations in A, t^A_{obs} the number of observed values in A, and t^A_{unobs} the number of unobserved values in A:

$$n(A) = \sum_{x_i \in A} n(x_i),$$
$$t^A_{obs} = |\{x_i \in A \mid n(x_i) > 0\}|, \quad t^A_{unobs} = |\{x_i \in A \mid n(x_i) = 0\}|.$$

Let $\underline{P}_{ANPI}$ denote the natural extension of $\mathcal{I}_{NPI}$ and m^{ANPI} the associated Möbius inverse. According to Theorem 2.3.1, $\underline{P}_{ANPI}$ is determined in the following way:

$$\underline{P}_{ANPI}(A) = \frac{n(A) - \min\left(t - |\overline{A}|, t^A_{obs}\right)}{N}, \quad \forall A \subseteq \{x_1, x_2, \dots, x_t\}. \qquad (7.8)$$

The following properties are remarkable about the A-NPI-M credal set given by Equation (2.69):

- **Sample Size:**

  It is easy to observe that the A-NPI-M intervals
  $I_{ANPI}(x_i) = \left[\max\left(\frac{n(x_i)-1}{N}, 0\right), \min\left(\frac{n(x_i)+1}{N}, 1\right)\right]$, with $1 \leqslant i \leqslant t$, are narrower as the sample size is higher. Indeed, it can be easily checked that the width of $I_{ANPI}(x_i)$ is equal to $\frac{1}{N}$ if $n(x_i) = 0$ or $n(x_i) = N$, and $\frac{2}{N}$ otherwise, $\forall i = 1, 2, \dots, t$. Moreover, it always holds that $\frac{n(x_i)}{N} \in I_{ANPI}(x_i)$, $\forall i = 1, 2, \dots, t$. Hence, as the sample size converges to infinity, the A-NPI-M converges to a classical probability distribution, estimated by relative frequencies. This also happens with the IDM (recall that the width of an IDM probability interval is always equal to $\frac{s}{N+s}$, s being the IDM parameter).

- **Extreme points of the credal set:** The set of extreme points of an IDM credal set is quite simple to obtain. In fact, as shown in Section 2.3.1, it is composed by only t extreme points, which are determined in a straightforward way. In contrast, the set of extreme points of an A-NPI-M credal set is much more complex to obtain. For determining such a set, we employ Algorithm 2, which lets us obtain the set of extreme points of the credal associated with a reachable set of probability intervals. Recall that such a procedure is recursive and uses an implicit tree search where each node corresponds to a partial probability distribution. At the root node, the partial probability distribution is the one associated with the lower probabilities. Under the A-NPI-M, such a partial probability distribution is determined by $p(x_i) = \max\left(0, \frac{n(x_i)-1}{N}\right), \quad \forall i = 1, 2, \ldots, t$. Each child node is a refinement of its parent node in which one component is incremented. The leaf nodes of the tree correspond to the extreme points. The algorithm maintains a global set $\mathrm{Ext}\left(\mathcal{P}\left(\mathcal{I}_{ANPI}\right)\right)$, which contains the extreme points of $\mathcal{P}\left(\mathcal{I}_{ANPI}\right)$ found in each moment. At each node, there are two local variables: Expl, which contains the indices i whose component cannot be incremented as $p\left(x_i\right) = \min\left(\frac{n(x_i)+1}{N}, 1\right), \quad i \in \{1, 2, \ldots, t\}$, and a real value λ, which indicates the remaining probability mass to distribute among the components ($\lambda = 1 - \sum_{i=1}^{t} p\left(x_i\right)$). Clearly, at the root node, $\mathrm{Expl} = \emptyset$ and $\lambda = 1 - \sum_{i=1}^{t} \max\left(\frac{n(x_i)-1}{N}, 0\right) = \frac{t_{obs}}{N}$.

Algorithm 11 presents the procedure to obtain the set of extreme points of $\mathcal{P}\left(\mathcal{I}_{ANPI}\right)$.

Algorithm 11: Procedure to compute the set of extreme points of an A-NPI-M credal set.

```
Procedure Determine extreme point of an A-NPI-M credal set(Observed frequencies in the sample $(n(x_1), n(x_2), \ldots, n(x_t))$)
$\mathrm{Ext}\left(\mathcal{P}\left(\mathcal{I}_{ANPI}\right)\right) = \emptyset$
$\mathrm{Expl} = \emptyset$
$\lambda = \frac{t_{obs}}{N}$
for $i = 1$ to $t$ do
    $p\left(x_i\right) = \max\left(0, \frac{n(x_i)-1}{N}\right)$
GetExtremePoints(p,$\lambda$,Expl)
for $i = 1$ to $t$ do
    if $i \notin \mathrm{Expl}$ then
        if $n(x_i) = 0$ or $n(x_i) = t$ then
            $aux \leftarrow \frac{1}{N}$
        else
            $aux \leftarrow \frac{2}{N}$
        if $\lambda \leqslant aux$ then
            $p' \leftarrow \left(p\left(x_1\right), \ldots, p\left(x_i\right) + \lambda, \ldots, p\left(x_t\right)\right)$
            if $p' \notin \mathrm{Ext}\left(\mathcal{P}\left(\mathcal{I}_{ANPI}\right)\right)$ then
                $\mathrm{Ext}\left(\mathcal{P}\left(\mathcal{I}_{ANPI}\right)\right) \leftarrow \mathrm{Ext}\left(\mathcal{P}\left(\mathcal{I}_{ANPI}\right)\right) \cup \{p'\}$
        else
            $p' \leftarrow \left(p\left(x_1\right), \ldots, \max\left(\frac{n(x_i)+1}{N}, 1\right), \ldots, p\left(x_t\right)\right)$
            GetExtremePoints($p'$, $\lambda - aux$, $\mathrm{Expl} \cup \{i\}$)
```

We show below an example of how to obtain the set of extreme points of an A-NPI-M credal set:

Example 7.3.1 *Suppose that* X *is a variable that takes values in* $\{x_1, x_2, x_3\}$, $n(x_1) = 15$, $n(x_2) = 10$, *and* $n(x_3) = 20$. *In this case,* $N = 45$, *and we have the following intervals associated with the A-NPI-M:*

$$\mathcal{I}_{ANPI} = \left\{ \left[\frac{14}{45}, \frac{16}{45}\right], \left[\frac{9}{45}, \frac{11}{45}\right], \left[\frac{19}{45}, \frac{21}{45}\right] \right\}.$$

Let $\mathcal{P}\left(\mathcal{I}_{ANPI}\right)$ *denote the credal set corresponding to* $\mathcal{I}_{ANPI}$. *If we apply Algorithm 11, we obtain the following 6 extreme points of* $\mathcal{P}\left(\mathcal{I}_{ANPI}\right)$*:*

$$p_1 = \left(\frac{16}{45}, \frac{10}{45}, \frac{19}{45}\right), \quad p_2 = \left(\frac{16}{45}, \frac{9}{45}, \frac{20}{45}\right), \quad p_3 = \left(\frac{15}{45}, \frac{11}{45}, \frac{19}{45}\right),$$
$$p_4 = \left(\frac{14}{45}, \frac{11}{45}, \frac{20}{45}\right), \quad p_5 = \left(\frac{15}{45}, \frac{9}{45}, \frac{21}{45}\right), \quad p_6 = \left(\frac{14}{45}, \frac{10}{45}, \frac{21}{45}\right),$$

where $p_i = \left(p_i\left(x_1\right), p_i\left(x_2\right), p_i\left(x_3\right)\right), \quad \forall i = 1, 2 \ldots, 6.$

- **Credal set associated with the IDM:**

  As said previously, the IDM strongly depends on the s parameter. IDM intervals are broader as the s value is higher. The value of the s parameter in IDM intervals indicates the previous knowledge assumed about the data; the higher is the s value, the more imprecise is assumed to be the data. This dependence is quite notable when the sample size is small, as happens at leaf nodes of Decision Trees for classification. Indeed, for a fixed s value, IDM intervals are wider as N is lower.

  In contrast, A-NPI-M intervals have no dependence on any parameter; they only depend on the relative frequencies and the sample size. For $s = 1$, one of the values recommended in [209], and the most used in practical applications, IDM intervals are contained in A-NPI-M intervals:

$$0 \leqslant \frac{n(x_j)}{N+1},$$

$$n(x_j) \leqslant N+1 \Rightarrow Nn(x_j) - 1 - N + n(x_j) \leqslant Nn(x_j) \Rightarrow \frac{n(x_j)-1}{N} \leqslant \frac{n(x_j)}{N+1},$$

$$1 \geqslant \frac{n(x_j)+1}{N+1}, \qquad \frac{n(x_j)+1}{N} \geqslant \frac{n(x_j)+1}{N+1}.$$

In this way,

$$\left[\frac{n(x_j)}{N+1}, \frac{n(x_j)+1}{N+1}\right] \subseteq \left[\max\left(\frac{n(x_j)-1}{N}, 0\right), \min\left(\frac{n(x_j)+1}{N}, 1\right)\right],$$
$$\forall j = 1, 2, \ldots, K.$$

Thus, the A-NPI-M is a more imprecise model than the IDM with $s = 1$.

- **Möbius Inverse:**

  In [59], it was shown that, for a specific configuration of the probability wheel in the NPI-M, the associated Möbius inverse is non-negative. Nonetheless, a set of A-NPI-M probability intervals cannot always be expressed by a belief function. The contrary happens with a set of IDM intervals, as pointed out in Section 2.3.1. The following example shows that the Möbius inverse corresponding to the A-NPI-M can have negative values.

Example 7.3.2 *Suppose that we have a discrete variable* X *and that* $\{x_1, x_2, x_3, x_4\}$ *are its possible values. Let us assume that* $n(x_j) > 0 \quad \forall j = 1, 2, 3, 4, \quad N = n(x_1) + n(x_2) + n(x_3) + n(x_4)$. *It holds that:*[3]

$$\underline{P}_{ANPI}\left(\{x_j\}\right) = \frac{n(x_j)-1}{N} = m^{ANPI}\left(\{x_j\}\right), \quad \forall j = 1, 2, 3, 4,$$

$$\underline{P}_{ANPI}\left(\{x_i, x_j\}\right) = \frac{n(x_i)+n(x_j)-2}{N}, \quad \forall 1 \leqslant i < j \leqslant 4,$$

$$\begin{aligned} m^{ANPI}\left(\{x_i, x_j\}\right) =& \underline{P}_{ANPI}\left(\{x_i, x_j\}\right) - \underline{P}_{ANPI}(\{x_i\}) - \\ & \underline{P}_{ANPI}\left(\{x_j\}\right) = 0, \quad \forall 1 \leqslant i < j \leqslant 4, \end{aligned}$$

$$\underline{P}_{ANPI}\left(\{x_i, x_j, x_k\}\right) = \frac{n(x_i)+n(x_j)+n(x_k)-1}{N}, \quad \forall 1 \leqslant i < j < k \leqslant 4,$$

$$\begin{aligned} & m^{ANPI}\left(\{x_i, x_j, x_k\}\right) = \underline{P}_{ANPI}\left(\{x_i, x_j, x_k\}\right) - \underline{P}_{ANPI}\left(\{x_i, x_j\}\right) - \\ & \underline{P}_{ANPI}(\{x_i, x_k\}) - \underline{P}_{ANPI}\left(\{x_j, x_k\}\right) + \underline{P}_{ANPI}(\{x_i\}) + \\ & \underline{P}_{ANPI}\left(\{x_j\}\right) + \underline{P}_{ANPI}(\{x_k\}) = \frac{2}{N}, \quad \forall 1 \leqslant i < j < k \leqslant 4, \end{aligned}$$

$$m^{ANPI}(\{x_1, x_2, x_3, x_4\}) = 1 - \sum_{1 \leqslant i < j < k \leqslant 4} m^{ANPI}\left(\{x_i, x_j, x_k\}\right) -$$

$$\sum_{1 \leqslant i < j \leqslant 4} m^{ANPI}\left(\{x_i, x_j\}\right) - \sum_{j=1}^{4} m^{ANPI}\left(\{x_j\}\right) = -\frac{4}{N} < 0.$$

[3] Within this example, we use Theorem 2.3.1 to compute $\underline{P}_{ANPI}$.

In addition, if we use Theorem 7.2.3 in the previous example, we have that $X_0 = X_1 = \emptyset$, $X_2 = X$, $\sum_{x_i \in X_2} u_i > 1 \Rightarrow 0 = \sum_{x_i \in X_1} l_i > 1 - \sum_{x_i \in X_2} u_i$, and

$$\begin{aligned} u_i + u_j &= 1 - \sum_{k=1, k \neq i, k \neq j}^{4} l_k, \quad \forall x_i, x_j \in X_2 \setminus X_0, \\ \sum_{x_i \in X_2 \setminus X_0} u_i &= 1 + \frac{4}{N} < 1 + \frac{8}{N} = \frac{4}{N}(4-1) + 1 - \frac{4}{N} \\ &= (1-L)\left(|X_2 \setminus X_0| - 1\right) + \sum_{x_i \in X_2 \setminus X_0} l_i. \end{aligned}$$

Therefore, the coherent lower probability function associated with an A-NPI-M credal set is not always a belief function, and, consequently, A-NPI-M credal sets do not belong to Evidence theory.

Clearly, for singletons, the Möbius inverse corresponding to the A-NPI-M coincides with the A-NPI-M lower probability, i.e,

$$m^{ANPI}\left(\{x_j\}\right) = \underline{P}_{ANPI}\left(\{x_j\}\right) = \max\left(\frac{n(x_j) - 1}{N}, 0\right), \quad \forall j = 1, 2, \ldots, t.$$

Let us analyze some properties of Möbius inverse associated with the A-NPI-M for sets of cardinality greater or equal than 2.

In the following result, we show that the Möbius inverse for a set whose cardinality is greater or equal than 2 does not depend on the observed frequencies of the values in the set.

Proposition 7.3.1 *If* $A \subseteq \{x_1, \ldots, x_t\}$ *with* $|A| \geqslant 2$*, then* $m^{ANPI}(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B| + 1} \frac{\min\left(t^B_{obs}, |\overline{B}|\right)}{N}$.

Proof: In order to determine the number of subsets of A with a certain cardinality that contain x_i, with $1 \leqslant i \leqslant t$, we think in the same way as in the proof of Proposition 7.2.1. Hence,

$$
\begin{aligned}
m^{ANPI}(A) &= \sum_{B \subseteq A} (-1)^{|A \setminus B|} \underline{P}_{ANPI}(B) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \frac{n(B) - \min\left(t^B_{obs}, |\overline{B}|\right)}{N} \\
&= \frac{1}{N} \times \left[\sum_{B \subseteq A} (-1)^{|A \setminus B|} \sum_{x_j \in B} n(x_j) - \sum_{B \subseteq A} (-1)^{|A \setminus B|} \min\left(t^B_{obs}, |\overline{B}|\right) \right] \\
&= \frac{1}{N} \times \left[\sum_{x_j \in A} n(x_j) \times \left(\sum_{i=0}^{|A|-1} (-1)^i \binom{|A|-1}{i} \right) - \right. \\
&\left. \sum_{B \subseteq A} (-1)^{|A \setminus B|} \min\left(t^B_{obs}, |\overline{B}|\right) \right] \\
&\sum_{B \subseteq A} (-1)^{|A \setminus B|+1} \frac{\min\left(t^B_{obs}, |\overline{B}|\right)}{N}.
\end{aligned}
$$

We have used Lemma 7.2.1 in the last equality since, by hypothesis, $|A| - 1 \geqslant 1$.

□

The following proposition is very useful to simplify the calculation of the Möbius inverse:

Proposition 7.3.2 *It holds that*

$$
\sum_{B \subseteq A} (-1)^{|A \setminus B|+1} t^B_{obs} = 0, \quad \forall A \subseteq \{x_1, x_2, \dots, x_t\} : |A| \geqslant 2.
$$

Proof: If $|A| = t^A_{obs}$, then the result is obtained as in the proof of Proposition 7.3.1.

Suppose that $t^A_{obs} < |A|$. For determining the number of subsets of A that have i observed values, with $0 \leqslant i \leqslant t^A_{obs}$, we think in the following way: We can choose the i values between the t^A_{obs} observed ones in $\binom{t^A_{obs}}{i}$ possible ways. Now, the set A has $|A| - t^A_{obs}$ non-observed values. Therefore, a subset of A that has i observed values has j non-observed values, with $0 \leqslant j \leqslant |A| - t^A_{obs}$.

The j non-observed values can be chosen in $\binom{|A|-t^A_{obs}}{j}$ possible ways. Clearly, the cardinality of the corresponding subset is $i+j$. Hence:

$$\sum_{B\subseteq A}(-1)^{|A\setminus B|+1}\,t^B_{obs} =$$

$$\sum_{i=1}^{t^A_{obs}} i\times\binom{t^A_{obs}}{i}\times\left[\sum_{j=0}^{|A|-t^A_{obs}}\binom{|A|-t^A_{obs}}{j}(-1)^{|A|-i-j+1}\right] =$$

$$\sum_{i=1}^{t^A_{obs}} i\times\binom{t^A_{obs}}{i}\times(-1)^{|A|-i+1}\times\left[\sum_{j=0}^{|A|-t^A_{obs}}\binom{|A|-t^A_{obs}}{j}(-1)^{j}\right] =$$

$$\sum_{i=1}^{t^A_{obs}} i\times\binom{t^A_{obs}}{i}\times(-1)^{|A|-i+1}\times 0 = 0.$$

The penultimate equality is due to Lemma 7.2.1 because $|A|-t^A_{obs}\geqslant 1$.

□

The following corollary allows us to know a condition under which the Möbius inverse for a set of cardinality greater or equal than 2 is equal to 0: the number of observed values in the set is not greater than the cardinality of the complementary set.

Corollary 7.3.1 *If* $A\subseteq\{x_1,x_2,\dots,x_t\}$ *with* $|A|\geqslant 2$ *satisfies* $t^A_{obs}\leqslant|\overline{A}|$*, then* $m^{ANPI}(A)=0$.

Proof: According to Proposition 7.3.1,

$$m^{ANPI}(A)=\sum_{B\subseteq A}(-1)^{|A\setminus B|+1}\frac{\min\left(t^B_{obs},|\overline{B}|\right)}{N}.$$

Now, under our hypothesis, if $B\subseteq A$, then $t^B_{obs}\leqslant t^A_{obs}\leqslant|\overline{A}|\leqslant|\overline{B}|$. In consequence, $\min\left(t^B_{obs},|\overline{B}|\right)=t^B_{obs}\quad\forall B\subseteq A$.

Proposition 7.3.2 lets us conclude that:

$$m^{ANPI}(A)=\frac{1}{N}\times\sum_{B\subseteq A}(-1)^{|A\setminus B|+1}\,t^B_{obs}=0.$$

□

Thus, if the cardinality of a set is greater than 1 but lower or equal than the half of the number of possible values of X, then the Möbius inverse for that set is equal to 0. It is expressed in the following result:

Corollary 7.3.2 *If* $A \subseteq \{x_1, x_2, \ldots, x_t\}$ *with* $2 \leqslant |A| \leqslant \frac{t}{2}$*, then* $m^{ANPI}(A) = 0$.

Proof: Under our hypotesis, it is obvious that $t_{obs}^{A} \leqslant |A| \leqslant \frac{t}{2} \leqslant |\overline{A}|$, and, from Corollary 7.3.1, it is concluded that $m^{ANPI}(A) = 0$. □

The following proposition lets us obtain a simpler expression for the calculation of the Möbius inverse for sets of cardinality greater than 1.

Proposition 7.3.3 $\forall A \subseteq \{x_1, \ldots, x_t\}$ *with* $|A| \geqslant 2$*, it holds that* $m^{ANPI} = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \frac{\max(t_{obs}^{B} - |\overline{B}|, 0)}{N}$.

Proof: Due to Propositions 7.3.1 and 7.3.2:

$$\begin{aligned}
&m^{ANPI}(A) = m^{ANPI}(A) - 0 = \\
&\sum_{B \subseteq A} (-1)^{|A \setminus B|+1} \frac{\min\left(t_{obs}^{B}, |\overline{B}|\right)}{N} - \sum_{B \subseteq A} (-1)^{|A \setminus B|+1} \frac{t_{obs}^{B}}{N} = \\
&\sum_{B \subseteq A} (-1)^{|A \setminus B|} \frac{t_{obs}^{B}}{N} - \sum_{B \subseteq A} (-1)^{|A \setminus B|} \frac{\min\left(t_{obs}^{B}, |\overline{B}|\right)}{N} = \\
&\sum_{B \subseteq A} (-1)^{|A \setminus B|} \frac{\max\left(t_{obs}^{B} - |\overline{B}|, 0\right)}{N}.
\end{aligned}$$

□

Finally, the following result allows us to obtain an expression of the Möbius inverse for a non-singleton subset as a function of the sample size, the number of values of X, the cardinality of the set, and the number of observed values in the set.

Proposition 7.3.4 *If* $A \subseteq \{x_1, \ldots, x_t\}$ *with* $|A| \geqslant 2$*, then*

$$\begin{aligned}
&m^{ANPI}(A) = \frac{1}{N} \times \sum_{i=1}^{t_{obs}^{A}} \binom{t_{obs}^{A}}{i} \times \\
&\left[\sum_{j=0}^{|A| - t_{obs}^{A}} \binom{|A| - t_{obs}^{A}}{j} \times (-1)^{|A|-i-j} \times \max(2i - t + j, 0) \right].
\end{aligned}$$

Proof: The result is obtained by applying Proposition 7.3.3 and thinking as in the first step of the proof of Proposition 7.3.2. □

Therefore, we can conclude that the calculation of the Mobius inverse for the A-NPI-M is far more complex than for the IDM.

7.3.1 A-NPI-M credal sets vs IDM credal sets

We can summarize the results of the comparison between credal sets corresponding to the A-NPI-M and IDM credal sets as follows:

1. Credal sets associated with the A-NPI-M converge to probability distributions estimated by means of relative frequencies as the sample size converges to infinity, as IDM credal sets.

2. The set of extreme points of an A-NPI-M credal set is much more complex to obtain than the set of extreme points of an IDM credal set.

3. The A-NPI-M is more imprecise than the IDM with the most utilized value of the s parameter.

4. A-NPI-M credal sets cannot always be represented via belief functions, unlike IDM credal sets.

5. The Möbius inverse is far more difficult to calculate with the A-NPI-M than with the IDM.

6. We must remark that, unlike the A-NPI-M, the IDM assumes previous knowledge about the data, strongly depending on a parameter.

7.4 Conclusions

Sometimes, a single probability distribution is not suitable for representing the probabilistic knowledge about a finite set or a discrete variable because the available information is not sufficient. For this reason, several imprecise probability theories and models have been developed in the literature. In this chapter, we have analyzed some relations between such theories and models.

On the one hand, belief functions and reachable probability intervals are imprecise probability theories that have been frequently used in practical applications to deal with uncertainty. It is known that, in general, belief functions are not generalizations of reachable probability intervals, and the converse is also not satisfied. In this chapter, we have described credal sets belonging to both belief functions and reachable probability intervals.

Specifically, we have given a set of necessary and sufficient conditions for a reachable set of probability intervals on a finite set to be representable by a belief function. For checking such conditions, it is needed to consider three subsets and test some simple inequalities with the sums of the lower and upper probabilities on these subsets. The computation of the subsets is also simple and fast.

A characterization of belief functions representable via reachable probability intervals has also been provided. In concrete, it has been demonstrated that a belief function can be represented by its associated set of belief intervals for singletons if, and only if, the difference between any pair of non-singleton focal elements of the corresponding BPA has a cardinality lower or equal than one. Using this condition, we have characterized some special types of belief functions, such as p-boxes or necessity measures, representable through reachable sets of probability intervals.

On the other hand, the Non-Parametric Predictive Inference Model (NPI-M) presents some advantages over the Imprecise Dirichlet Model (IDM) as its inferences often give more intuitively coherent results. Moreover, the NPI-M, unlike the IDM, does not assume previous knowledge about the data via a parameter. The Approximate Non-Parametric Predictive Inference Model (A-NPI-M) starts from the NPI-M and considers the convex hull of the set of probability distributions compatible with this model. In consequence, the A-NPI-M is easier to manage than the NPI-M since it considers a credal set associated with a reachable set of probability intervals. In this chapter, we have analyzed the main properties of credal sets associated with the A-NPI-M, comparing them with IDM credal sets.

We have shown that, as with the IDM, as long as the sample size converges to infinity, A-NPI-M credal sets converge to a single probability distribution, estimated by relative frequencies. It has been shown that the A-NPI-M is a more imprecise model than the IDM with the most used value of the s parameter, the one recommended in the literature. One of the most remarkable properties of A-NPI-M credal sets is that they cannot always be represented by a belief function, unlike IDM credal sets. The calculation of the Möbius inverse for the A-NPI-M is much more complex than for the IDM. The same occurs with the set of extreme points of the credal set. Thereby, the A-NPI-M is a notably more complex model than the IDM. However, we must remark that the IDM makes previous assumptions about the data through a parameter, unlike the A-NPI-M. As pointed out previously, it supposes a drawback when the IDM is used in classification because the parameter strongly influences the results, and it has not been possible so far to associate the optimal value of the parameter with each dataset [18].

8 ANALYSIS AND PROPOSALS OF UNCERTAINTY MEASURES ON IMPRECISE PROBABILITIES

8.1 Introduction

When imprecise probability theories and models arise, measures for quantifying the uncertainty-based information in such theories and models are required. As explained before, the origin of the study of uncertainty measures in imprecise probabilities resides in the study of uncertainty measures in Evidence theory (ET).

It has been shown that, so far, the only uncertainty measure in ET that satisfies all essential mathematical properties and behavioral requirements is the maximum entropy. However, the procedures developed so far to compute the maximum entropy are notably complex [7]. This supposes a drawback for using such an uncertainty measure in practical applications. For this reason, many alternatives to this measure have been proposed during the last years.

Among these alternatives, one of the most known is the Deng entropy [77]. The basis of this measure is that the uncertainty-based information is strongly influenced by the number of alternatives. Abellán [4] demonstrated that the Deng entropy does not verify most of the vital mathematical properties, and its behavior in many situations is undesirable. Hence, the Deng entropy should be cautiously used in practical applications.

A modification of the Deng entropy was introduced in [236]. Such a modification supposes an improvement over the original Deng entropy since it considers the total number of possible alternatives to a higher degree. Another version of the Deng entropy was proposed in [69]. It performs better than the original Deng entropy because it considers the intersections between statements on uncertainty. In this chapter, we demonstrate that these modifications of the Deng entropy do also not verify most of the essential mathematical properties. Moreover, we also show that, similar to the original Deng entropy, the behavior of the mentioned modifications in some situations is also questionable. Therefore, they should be cautiously utilized in practical applications, as happens with the original Deng entropy.

Many other alternatives to the maximum entropy in ET proposed during the last years are based on the belief intervals for singletons. It is known that, when using belief intervals for singletons instead of BPAs, some information might be lost. Nonetheless, belief intervals for singletons are easier to manage than BPAs for representing uncertainty-based information in ET because they let us quickly know the uncertain area associated with each alternative (see Figure 3.1). In this way, belief intervals for singletons are considered an interesting tool for quantifying uncertainty-based information in ET.

In this chapter, as a novelty, we carry out a study about the essential mathematical properties that have to be satisfied by every total uncertainty measure on belief intervals for singletons. We also analyze the crucial behavioral requirements for measures of this category. Our study is based on the one carried out by Abellán and Masegosa [21] for total uncertainty measures on BPAs. We study which of such fundamental mathematical properties and behaviors are satisfied by each one of the uncertainty measures on belief intervals for singletons proposed so far. It is demonstrated that none of such measures satisfies all required mathematical properties and behaviors.

Furthermore, we present an uncertainty measure on belief intervals for singletons consisting of the maximum entropy on the credal set compatible with this set of intervals. We demonstrate that, unlike the uncertainty measures on belief intervals for singletons proposed so far, our proposal satisfies all crucial mathematical properties and behavioral requirements for this type of measure, even though its computation is more complex. We also show that our proposed measure could be considered an approximation to the maximum entropy of the credal set associated with a BPA as it provides an upper bound. Moreover, the former measure is far easier to compute than the latter.

In addition to ET, uncertainty measures on credal sets have also been proposed. As pointed out before, the maximum entropy is the well-established uncertainty measures on credal sets because it satisfies all the required mathematical properties for this kind of measure. The most general imprecise probability theory for which a procedure to compute the maximum entropy has been proposed is Choquet capacities of order 2 [15]. Abellán [2] showed how to calculate the main uncertainty measures on IDM credal sets (the maximum entropy and the uncertainty measures that let us decompose the maximum entropy into two measures that quantify conflict and non-specificity). Remark that the NPI-M, unlike the IDM, does not make prior assumptions about the data via a parameter. A procedure to compute the maximum entropy with the NPI-M was presented in [5].

Remark that the A-NPI-M notably simplifies the NPI-M as it considers the convex hull of the set probability distributions compatible with the exact model. In this chapter, we also present algorithms that allow obtaining the main uncertainty measures on A-NPI-M credal sets. We show that these procedures are more simple and efficient than the ones developed in the literature for general imprecise probability theories. Hence, with the A-NPI-M, it is possible to efficiently quantify the uncertainty-based information.

The remainder of this chapter is arranged as follows: In Section 8.2, we make a critical analysis of the aforementioned modifications of the Deng entropy. Section 8.3 details our study about the required mathematical properties and behaviors for uncertainty measures on belief intervals for singletons. Our proposed uncertainty measure on belief intervals for singletons is introduced in Section 8.4. Section 8.5 describes our proposed procedures for computing the main uncertainty measures with the A-NPI-M. This chapter is concluded in Section 8.6.

Within this chapter, let $X = \{x_1, x_2, \dots, x_t\}$ be a finite set[1], with $|X| = t$. Let $\wp(X)$ denote the power set of X and $\mathcal{P}(X)$ the set of all probability distributions on X.

8.2 Critique of the modifications of the Deng entropy

A modification of the Deng entropy known as the Zhou entropy was proposed in [236]. For a given BPA m on X, it is defined in the following way (See Equation (3.13)):

$$E_{Zhou}(m) = -\sum_{A \in \wp(X)} m(A) \log_2 \left(\frac{m(A)}{2^{|A|} - 1} \exp\left(\frac{|A| - 1}{|X|} \right) \right).$$

This function can be re-written as follows:

$$\begin{aligned} E_{Zhou}(m) = &\sum_{A \in \wp(X)} m(A) \log_2 (2^{|A|} - 1) - \\ &\sum_{A \in \wp(X)} m(A) \log_2 \left(\exp\left(\frac{|A| - 1}{|X|} \right) \right) - \sum_{A \in \wp(X)} m(A) \log_2 m(A). \end{aligned} \tag{8.1}$$

It could be considered that the first two terms in Equation (8.1) quantify the non-specificity part in a BPA since both of them are equal to 0 when m is a probability distribution. The third one might measure the conflict part, which coincides with the conflict value in the original Deng entropy. It can be observed that E_{Zhou} is also based on the idea of the Deng entropy as it gives

[1] Or a discrete variable that takes values in $\{x_1, x_2, \dots, x_t\}$.

a higher total uncertainty value when there are more alternatives. However, the increase is more controlled with this modification due to the second term.

Cui, Liu, Zhang, and Kang [69] proposed a new version of the Deng entropy that takes the intersections between the focal elements into account. For a given BPA m on X, it is defined as follows (See Equation (3.14)):

$$E_{Cui}(m) = -\sum_{A \in \wp(X)} m(A) \log_2 \left[\left(\frac{m(A)}{2^{|A|}-1} \right) \exp \left(\sum_{B \neq A \wedge m(B)>0} \frac{|A \cap B|}{2^{|X|-1}} \right) \right].$$

It is possible to re-write this function as follows:

$$\begin{aligned} E_{Cui}(m) = \sum_{A \in \wp(X)} m(A) \log_2 \left(2^{|A|}-1\right) - \sum_{A \in \wp(X)} m(A) \\ \log_2 \left[m(A) \times \exp \left(\sum_{B \neq A \wedge m(B)>0} \frac{|A \cap B|}{2^{|X|-1}} \right) \right]. \end{aligned} \tag{8.2}$$

In the above expression, the first term indicates non-specificity, while the second one captures conflict. Indeed, when m is a probability distribution, the first term is equal to 0, and the second one collapses to the Shannon entropy. Hence, in E_{Cui}, the non-specificity part coincides with the non-specificity part in the Deng entropy. However, the conflict part of E_{Cui} is lower than the conflict part of the Deng entropy due to the exponential term.

8.2.1 Mathematical properties of the modified Deng entropies

We show below which of the required mathematical properties for total uncertainty measures in ET, exposed in Section 3.3.1, are satisfied by the modified Deng entropies E_{Zhou} and E_{Cui}.

- **Probabilistic Consistency**: It is easy to observe that, if m is a probability distribution, then

  $$\sum_{A \in \wp(X)} m(A) \log_2 (2^{|A|}-1) = \sum_{A \in \wp(X)} m(A) \log_2 \left(\exp \left(\frac{|A|-1}{|X|} \right) \right) = 0,$$

  and, thus, E_{Zhou} collapses to the Shannon entropy.

  We may also note that, in this case, it holds that:

  $$\exp \left(\sum_{B \neq \{x_i\} \wedge m(B)>0} \frac{|\{x_i\} \cap B|}{2^{|X|-1}} \right) = 1, \quad \forall i = 1, 2, \ldots, t,$$

and we deduce that E_{Cui} also coincides with the Shannon entropy for probability distributions. Consequently, both modifications of the Deng entropy satisfy Probabilistic Consistency.

- **Set Consistency**: If m is a BPA on X such that $m(A) = 1$ for some $A \subseteq X$, then:

$$E_{Zhou}(m) = \log_2 (2^{|A|} - 1) - \log_2 \left(\exp\left(\frac{|A|-1}{|X|}\right)\right).$$

The following result shows that, in this case, if there are three or more alternatives, then the value of $E_{Zhou}(m)$ is strictly greater than the one obtained by the generalized Hartley measure. [2].

Proposition 8.2.1 *If* $|X| \geqslant 3$ *and* $m(A) = 1$ *for some* $A \subseteq X$, *then*

$$\log_2 (2^{|A|} - 1) - \log_2 \left(\exp\left(\frac{|A|-1}{|X|}\right)\right) > \log_2 (|A|), \quad \forall A \subseteq X, |A| \geqslant 2.$$

Proof: It is easy to check that:

$$\log_2 (2^{|A|} - 1) - \log_2 \left(\exp\left(\frac{|A|-1}{|X|}\right)\right) > \log_2 (|A|) \Leftrightarrow$$
$$\log_2 \left(\frac{2^{|A|}-1}{|A|}\right) > \log_2 \left(\exp\left(\frac{|A|-1}{|X|}\right)\right) \Leftrightarrow$$
$$\frac{2^{|A|}-1}{|A|} > \exp\left(\frac{|A|-1}{|X|}\right).$$

We distinguish 3 cases:

1. $|A| = 2$. In this situation:

$$\frac{2^{|A|}-1}{|A|} = \frac{3}{2} > 1.3956 = \exp\left(\frac{1}{3}\right) \geqslant \exp\left(\frac{|A|-1}{|X|}\right).$$

2. $|A| = 3$. Then:

$$\frac{2^{|A|}-1}{|A|} = \frac{7}{3} > 1.9477 = \exp\left(\frac{2}{3}\right) \geqslant \exp\left(\frac{|A|-1}{|X|}\right).$$

3. $|A| \geqslant 4$. In this case, since the function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = \frac{2^x-1}{x}$, $\forall x \in \mathbb{R}$, is clearly increasing, we have that:

$$\frac{2^{|A|}-1}{|A|} \geqslant \frac{2^4-1}{4} = \frac{15}{4} > \exp(1) > \exp\left(\frac{|A|-1}{|X|}\right).$$

[2] except for when A is a singleton, but, in that case, there is no uncertainty.

□

Therefore, E_{Zhou} does not satisfy the Set Consistency property.

Moreover, it is convenient to remark that there is a unique case where the non-specificity value of E_{Zhou} is lower than the non-specificity value of the Hartley measure: $|A| = |X| = 2$. Then,

$$E_{Zhou}(m) = \log_2(3) - \log_2\left(\exp\left(\frac{1}{2}\right)\right) = 0.8636 < 1 = \log_2(|A|).$$

Regarding E_{Cui}, when $m(A) = 1$ for some $A \subseteq X$,
$E_{Cui}(m) = \log_2\left(2^{|A|} - 1\right)$, as the original Deng entropy. What is more, E_{Cui} coincides with the Deng entropy when all focal elements are disjunct. In consequence, E_{Cui} neither satisfies Set Consistency, although, in these cases, E_{Cui} always provides a greater value than the Hartley measure, unlike E_{Zhou}.

- **Range**: If $|X| = 4$ and m is a BPA on X such that $m(X) = 1$, then:

$$\begin{aligned} E_{Zhou}(m) &= \log_2\left(2^{|X|} - 1\right) - \log_2\left(\exp\left(\frac{|X|-1}{|X|}\right)\right) \\ &= \log_2(15) - \log_2\left(\exp\left(\frac{3}{4}\right)\right) \\ &= 2.8249 > 2 = \log_2(4) = \log_2(|X|). \end{aligned}$$

In such a case,

$$E_{Cui}(m) = \log_2\left(2^{|X|} - 1\right) = \log_2(15) > \log_2(4) = \log_2(|X|).$$

Hence, the range property is not verified by E_{Zhou} nor E_{Cui}.

According to the results proved in [125, 237], the maximum value of the Deng entropy is equal to $\log_2\left(\sum_{A\subseteq X}\left(2^{|A|} - 1\right)\right)$. It is attained with the following BPA:

$$m^*(A) = \frac{2^{|A|} - 1}{\sum_{B\subseteq X}\left(2^{|B|} - 1\right)}, \quad \forall A \subseteq X.$$

It is easy to observe that, in this case, the value obtained by E_{Zhou} is lower than the one attained by the Deng entropy due to the second term of Equation (8.1). Likewise, in this situation, because of the exponential term of Equation (3.14), the value obtained by E_{Cui} is lower than the one obtained by the Deng entropy. Consequently, the ranges of E_{Zhou} and E_{Cui} are lower than the range of the original Deng entropy.

- **Subadditivity**: The following example shows that the modifications of the Deng entropy considered here are not subadditive:

Example 8.2.1 *Let us consider the finite sets* $X = \{x_1, x_2, x_3\}$ *and* $Y = \{y_1, y_2\}$*. Let* m *be the following BPA on the product space* $X \times Y$*:*

$$m(\{z_{11}, z_{12}, z_{21}\}) = 0.6, \quad m(\{z_{31}, z_{32}\}) = 0.1, \quad m(X \times Y) = 0.3,$$

where we have denoted $z_{ij} = (x_i, y_j)$*.*

Let $m^{\downarrow X}$ *and* $m^{\downarrow Y}$ *denote the marginal BPAs of* m *on* X *and* Y*, respectively. They are determined as follows:*

$$m^{\downarrow X}(\{x_1, x_2\}) = 0.6, \quad m^{\downarrow X}(\{x_3\}) = 0.1, \quad m^{\downarrow X}(X) = 0.3;$$

$$m^{\downarrow Y}(Y) = 1.$$

E_{Zhou} *takes the following values:*

$$E_{Zhou}(m) = 4.2583, \quad E_{Zhou}(m^1) = 2.5116, \quad E_{Zhou}(m^2) = 0.8636$$

It holds that $E_{Zhou}(m^1) + E_{Zhou}(m^2) = 3.3752$ *and, thus,* $E_{Zhou}(m^1) + E_{Zhou}(m^2) < E_{Zhou}(m)$*.*

Concerning E_{Cui}*:*

$$E_{Cui}(m) = 4.8674, \quad E_{Cui}(m^{\downarrow X}) = 1.4574, \quad E_{Cui}(m^{\downarrow Y}) = 1.585$$

Hence, $E_{Cui}(m^{\downarrow X}) + E_{Cui}(m^{\downarrow Y}) = 3.0424 < 4.8674 = E_{Cui}(m)$*.*

- **Additivity**: We show in the example below that E_{Zhou} and E_{Cui} do not verify the additivity property.

Example 8.2.2 *Let* $X = \{x_1, x_2, x_3\}$ *and* $Y = \{y_1, y_2\}$ *be two finite sets. Let* m^X *and* m^Y *be the following BPAs on* X *and* Y*, respectively:*

$$m^X(\{x_1, x_2\}) = 0.6, \quad m^X(\{x_3\}) = 0.1, \quad m^X(X) = 0.3;$$

$$m^Y(Y) = 1.$$

We consider now the BPA $m = m^X \times m^Y$ *on the product space* $X \times Y$*. It has the following values:*

$$m(\{z_{11}, z_{12}, z_{21}, z_{22}\}) = 0.6, \quad m(\{z_{31}, z_{32}\}) = 0.1, \quad m(X \times Y) = 0.3,$$

where, again, we have denoted $z_{ij} = (x_i, y_j)$*. It is easy to check that the marginal BPAs of* m *on* X *and* Y *are, respectively,* m^X *and* m^Y*, and they are non-interactive. We have the following values for* E_{Zhou} *and* E_{Cui}*:*

$$E_{Zhou}(m) = 4.7873, \quad E_{Zhou}(m^X) = 2.5116, \quad E_{Zhou}(m^Y) = 0.8636.$$

$$E_{Cui}(m) = 5.5138, \quad E_{Cui}(m^X) = 1.4574, \quad E_{Cui}(m^Y) = 1.585.$$

Thus, $E_{Zhou}(m^X) + E_{Zhou}(m^Y) = 3.3752 \neq 4.7873 = E_{Zhou}(m)$*, and* $E_{Cui}(m^X) + E_{Cui}(m^Y) = 3.0424 \neq 5.5138 = E_{Cui}(m)$*.*

- **Monotonicity**: The following example demonstrates that an increase or decrease of uncertainty-based information is not always coherently reflected by E_{Zhou}:

Example 8.2.3 *Let* $X = \{x_1, x_2\}$ *be a finite set and* m^1 *and* m^2 *the following BPAs on* X*:*

$$m^1(X) = 1;$$

$$m^2(\{x_1\}) = m^2(X) = 0.5.$$

We have the following values for E_{Zhou}*:*

$$E_{Zhou}(m^1) = 0.8636, \quad E_{Zhou}(m^2) = 1.4318.$$

Clearly, the information provided by m^2 *is greater than the one represented via* m^1 *(*m^1 *corresponds to total ignorance). Nevertheless,*
$E_{Zhou}(m_1) < E_{Zhou}(m_2)$*.*

In the following example, it is shown that the monotonicity requirement is also not verified by E_{Cui}.

Example 8.2.4 *Let us consider the finite set* $X = \{x_1, x_2\}$ *and the following BPAs on* X*:*

$$m^1(X) = 0.9, \quad m^1(\{x_1\}) = 0.1;$$

$$m^2(\{x_1\}) = m^2(\{x_2\}) = m^2(X) = \frac{1}{3}.$$

It is easy to observe that m^2 *represents more uncertainty-based information than* m^1*. However,* $E_{Cui}(m^1) = 1.4146 < 1.4721 = E_{Cui}(m^2)$*.*

In this way, among the required mathematical properties for uncertainty measures in ET, E_{Zhou} and E_{Cui} only verify Probabilistic Consistency. Most of such properties are crucial: If a BPA is defined on a product space, then the sum of the uncertainties in the marginal BPAs cannot be lower than the uncertainty involved in the original BPA; if we join two non-interactive BPAs, then the total amount of uncertainty-based information must not vary; when an increase of the uncertainty-based information contained in a BPA is produced, it does not make sense that the uncertainty increases. The modifications of the Deng entropy considered here present the mentioned shortcomings. The same happens with the original Deng entropy [4].

8.2.2 Some undesirable behaviors of the modifications of the Deng entropy

The original Deng entropy provides incoherent results in some situations because it does not consider the number of possible alternatives suitably [236]. The modified Deng entropy E_{Zhou} was proposed to solve this problem. Also, E_{Cui} improves the original Deng entropy because it considers the intersections between the focal elements. However, as we show in this subsection, both E_{Zhou} and E_{Cui} also present some behavioral drawbacks, as the original Deng entropy.

– Firstly, the maximum value of E_{Zhou} is not attained with the BPA associated with total ignorance, as we have observed in Example 8.2.3. This is an illogical situation because total ignorance implies total lack of information. This also happens with the Deng entropy [125, 237]. In general, since E_{Zhou} does not satisfy the monotonicity property, it is not always consistent with an increase or decrease of information, which is quite undesirable. E_{Cui} neither satisfies the monotonicity property. For this reason, it also obtains incoherent results in some scenarios, as in Example 8.2.4.

– As happens with the original Deng entropy, the range of the non-specificity part of E_{Zhou} is greater than the range of the conflict part, although the difference is not as great as with the original Deng entropy. The difference between both ranges increases as the number of possible alternatives is greater. The same occurs with E_{Cui}. In consequence, the conflict part in both modifications of the Deng entropy might have little importance when there are many alternatives. It could make sense as the main difference between uncertainty in ET and probability theory resides in the non-specificity part. Nonetheless, it is questionable and not coherent with the thoughts in the literature that both types of uncertainty in ET have the same weight.

– We should remark that, in the original Deng entropy, when the information is focused on a single set, the non-specificity value is always greater than the non-specificity value of the Hartley measure. The same happens with E_{Cui}. In fact, Deng entropy and E_{Cui} obtain identical values when there is a single focal element. When there two are possible alternatives, i.e, $X = \{x_1, x_2\}$, and a BPA m on X such that $m(X) = 1$, the value of E_{Zhou} is lower than the value of the Hartley measure, as shown in Section 8.2.1. We have also demonstrated that, in the rest of the cases where there is only one focal element, the value of E_{Zhou} is strictly greater than the value of the Hartley measure. It might be an inconsistent behavior.

– Regarding the conflict parts of E_{Zhou} and E_{Cui}, they can have positive values in cases in which all focal elements share an element. It is not logical since the conflict in ET corresponds to cases where the information is focused on sets whose intersection is empty. It also occurs with the original Deng entropy [4].

– Finally, the extension of the modifications the Deng entropy considered here to more general theories than ET is still an open question. As pointed out in Section 3.3.1 (RB4), it must be possible to extend an uncertainty measure in ET to more general theories.

8.3 Requirements for uncertainty measures on belief intervals for singletons

Let m be a BPA on X, Bel_m its associated belief function and Pl_m its corresponding plausibility function. Let $\mathcal{I}_m$ denote the set of belief intervals for singletons associated with m, computed by means of Equation (2.41).

We consider the following issues for a total uncertainty measure on $\mathcal{I}_m$:

- When there is a unique probability distribution compatible with this set of intervals, which occurs if, and only if, $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) \quad \forall i = 1, 2, \dots, t$, a total uncertainty measure on $\mathcal{I}_m$ has to coincide with the well-established uncertainty measure in probability theory, i.e, the Shannon entropy.

- If it is only known that the information is focused on a single subset $A \subseteq X$ with $|A| \geqslant 2$, that is, $Bel_m(\{x_i\}) = 0 \quad \forall i = 1, 2, \dots, t$, $Pl_m(\{x_i\}) =$

$0 \quad \forall x_i \notin A$ and $Pl_m(\{x_i\}) = 1 \quad \forall x_i \in A$, then a total uncertainty measure on $\mathcal{I}_m$ may have to coincide with the one established as appropriate in classical possibility theory. Nevertheless, as pointed by Wang and Song [212], it should be considered that the uncertainty in a classical set depends on its cardinality. Consequently, in these cases, it is only crucial that a total uncertainty measure on $\mathcal{I}_m$ is an increasing function of $|A|$.

- In the study carried out by Abellán and Masegosa [21], it was established that the range of a total uncertainty measure on BPAs has to be equal to $[0, \log_2 |X|]$, as in probability theory. However, this point is debatable since in ET there are more kinds of uncertainty than in probability theory and, thus, arguments for a larger range might be reasonable.

  Nonetheless, a total uncertainty measure on $\mathcal{I}_m$ must be non-negative. The value 0 must be reached if, and only if, the information is focused on a singleton, i.e $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$ for some $i \in \{1, \ldots, t\}$ and $Bel_m\left(\{x_j\}\right) = Pl_m\left(\{x_j\}\right) = 0 \quad \forall j \in \{1, 2, \ldots, t\} \setminus \{i\}$. It can be stated that it is the only case in which there is no uncertainty. Furthermore, where there is an absolute lack of information, i.e, when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1 \quad \forall i = 1, 2, \ldots, t$, a total uncertainty measure on $\mathcal{I}_m$ must attain its maximum value.

- As happens with BPAs, a total uncertainty measure on $\mathcal{I}_m$ must be consistent when an increase or decrease of information is produced. In terms of belief intervals for singletons, the set of belief intervals for singletons associated with a BPA m_1,
  $\mathcal{I}_{m_1} = \{[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})], \quad i = 1, \ldots, t\}$, involves more uncertainty-based information than the one corresponding to another BPA m_2, $\quad \mathcal{I}_{m_2} = \{[Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad i = 1, \ldots, t\}$, if

  $$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1, 2, \ldots, t. \tag{8.3}$$

  In the following proposition, we show that the condition given in Equation (8.3) is equivalent to the fact that the set of probability distributions consistent with $\mathcal{I}_{m_1}$ is contained in the set of probability distributions compatible with $\mathcal{I}_{m_2}$.

Proposition 8.3.1 *Let $\mathcal{P}\left(\mathcal{I}_{m_j}\right)$ denote the credal set consistent with $\mathcal{I}_{m_j}$, computed via Equation (2.42), for $j = 1, 2$. It holds that:*

$$\mathcal{P}(\mathcal{I}_{m_1}) \subseteq \mathcal{P}(\mathcal{I}_{m_2}) \Leftrightarrow$$
$$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1, 2, \ldots, t.$$

Proof: Suppose that $Bel_{m_1}(\{x_i\}) < Bel_{m_2}(\{x_i\})$, for some $i \in \{1, 2, \ldots, t\}$. As $\mathcal{I}_{m_1}$ is reachable, $\exists p_i \in \mathcal{P}(\mathcal{I}_{m_1})$ such that $p_i(x_i) = Bel_{m_1}(\{x_i\}) < Bel_{m_2}(\{x_i\})$, which implies that $p_i \notin \mathcal{P}(\mathcal{I}_{m_2})$.

Likewise, if $Pl_{m_1}(\{x_i\}) > Pl_{m_2}(\{x_i\})$, then $\exists p_i' \in \mathcal{P}(\mathcal{I}_{m_1})$ satisfying $p_i'(x_i) = Pl_{m_1}(\{x_i\}) > Pl_{m_2}(\{x_i\})$, and, thus, $p_i' \notin \mathcal{P}(\mathcal{I}_{m_2})$.

In consequence, if $\mathcal{P}(\mathcal{I}_{m_1}) \subseteq \mathcal{P}(\mathcal{I}_{m_2})$, then the condition given in Equation (8.3) must be verified.

Let us assume now that

$$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1, 2, \ldots, t.$$

If $p \in \mathcal{P}(\mathcal{I}_{m_1})$, then:

$$\begin{aligned} & Bel_{m_2}(\{x_i\}) \leqslant Bel_{m_1}(\{x_i\}) \leqslant p(x_i) \leqslant \\ & Pl_{m_1}(\{x_i\}) \leqslant Pl_{m_2}(\{x_i\}), \quad \forall i = 1, 2, \ldots, t. \end{aligned}$$

This implies that

$$Bel_{m_2}(\{x_i\}) \leqslant p(x_i) \leqslant Pl_{m_2}(\{x_i\}), \quad \forall i = 1, 2, \ldots, t,$$

and we conclude that $p \in \mathcal{P}(\mathcal{I}_{m_2})$.

□

Deng and Jiang [76] analyzed this requirement for total uncertainty measures on belief intervals for singletons by utilizing the criterion established to decide whether a certain BPA m_1 contains the uncertainty-based information involved by another BPA m_2: $Bel_{m_1}(A) \geqslant Bel_{m_2}(A)$ and $Pl_{m_1}(A) \leqslant Pl_{m_2}(A), \quad \forall A \subseteq X$. It should be noted that this condition is stronger than the one imposed in Equation (8.3).

- Suppose now that $X = \{x_1, \ldots, x_t\}$ and $Y = \{y_1, \ldots, y_{t'}\}$ are two finite sets. Let m be a BPA on the product space $X \times Y$ and Bel_m and Pl_m the belief and plausibility functions associated with m, respectively. Let us consider the set of belief intervals for singletons corresponding to m, $\mathcal{I}_m$. Let $\mathcal{P}(\mathcal{I}_m)$ denote the credal set associated with $\mathcal{I}_m$. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$ be the projections of $\mathcal{I}_m$ on X and Y, respectively, determined through Proposition 2.2.9.

  For a total uncertainty measure on the belief intervals for singletons associated with a BPA defined on a product space, it is important that, when it is projected on the marginal sets, the total uncertainty does not decrease. This is related to the subadditivity property for total uncertainty measures on BPAs. Nevertheless, if a total uncertainty measure is based on belief intervals for singletons, it is much more coherent that this requirement is imposed through the projections of such intervals rather the marginal BPAs. We show with an example below that the belief intervals for singletons corresponding to the marginal BPAs do not always coincide with the marginalization of the belief intervals for singletons.

Example 8.3.1 *Let us consider the finite sets* $X = \{x_1, x_2, x_3\}$ *and* $Y = \{y_1, y_2\}$. *Let* m *be the BPA on* $X \times Y$ *given by:*

$$m\left(\{z_{11}, z_{12}, z_{21}\}\right) = 0.7, \quad m\left(\{z_{31}, z_{32}\}\right) = 0.1,$$

$$m\left(\{z_{11}, z_{12}, z_{21}, z_{22}, z_{31}, z_{32}\}\right) = 0.2,$$

where $z_{ij} = (x_i, y_j)$, *for* $i = 1, 2, 3, \quad j = 1, 2.$

For singletons, we have the following belief intervals:

$$z_{11} \to \left[Bel_m\left(\{z_{11}\}\right), Pl_m\left(\{z_{11}\}\right)\right] = [0, 0.9];$$

$$z_{12} \to \left[Bel_m\left(\{z_{12}\}\right), Pl_m\left(\{z_{12}\}\right)\right] = [0, 0.9];$$

$$z_{21} \to \left[Bel_m\left(\{z_{21}\}\right), Pl_m\left(\{z_{21}\}\right)\right] = [0, 0.9];$$

$$z_{22} \to \left[Bel_m\left(\{z_{22}\}\right), Pl_m\left(\{z_{22}\}\right)\right] = [0, 0.2];$$

$$z_{31} \to \left[Bel_m\left(\{z_{31}\}\right), Pl_m\left(\{z_{31}\}\right)\right] = [0, 0.3];$$

$$z_{32} \to \left[Bel_m\left(\{z_{32}\}\right), Pl_m\left(\{z_{32}\}\right)\right] = [0, 0.3]$$

Let $m^{\downarrow X}$ *denote the marginal BPA of* m *on* X. *We have that:*

$$m^{\downarrow X}\left(\{x_1, x_2\}\right) = 0.7,$$

$$m^{\downarrow X}\left(\{x_3\}\right) = 0.1, \quad m^{\downarrow X}(X) = 0.2.$$

The belief intervals for singletons associated with m_X *are the following ones:*

$$x_1 \to [0, 0.9]; \quad x_2 \to [0, 0.9]; \quad x_3 \to [0.1, 0.3].$$

Nevertheless, the result of the projection of the belief intervals for singletons corresponding to m *on* X *is the following one:*

$$x_1 \to [0, 1]; \quad x_2 \to [0, 1]; \quad x_3 \to [0, 0.6].$$

Let $\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$ and $\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$ denote the credal sets corresponding to $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$, respectively. If these credal sets are independent, then the value of a total uncertainty measure on $\mathcal{I}_m$ must coincide with the sum of the total uncertainty values on $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow y}$. This is associated with the additivity property for total uncertainty measures on BPAs, but, again, it makes more sense to consider the marginal belief intervals for singletons than the marginal BPAs.

As pointed out in Section 2.2.1.3, for independence of credal sets, the concept of strong independence is commonly used in the literature. There is strong independence under $\mathcal{P}(\mathcal{I}_m)$ if, and only if,

$$\mathcal{P}(\mathcal{I}_m) = \mathrm{CH}\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right), \tag{8.4}$$

where CH denotes the convex hull of a set of probability distributions.

Hence, a total uncertainty measure on $\mathcal{I}_m$, $\mathrm{TUM}(\mathcal{I}_m)$, must satisfy the following mathematical properties:[3]

1. **Probabilistic Consistency**: When $\mathrm{Bel}_m(\{x_i\}) = \mathrm{Pl}_m(\{x_i\})$ $\forall i = 1, 2, \ldots, t, \quad \mathrm{TUM}(\mathcal{I}_m)$ has to collapse to the Shannon entropy:

$$\mathrm{TUM}(\mathcal{I}_m) = -\sum_{i=1}^{t} \mathrm{Bel}_m(\{x_i\}) \log_2(\mathrm{Bel}_m(\{x_i\})).$$

2. **Generalized Set Consistency**: If $\exists A \subseteq X$ with $|A| \geqslant 2$ such that $\mathrm{Bel}_m(\{x_i\}) = 0 \quad \forall i = 1, \ldots, t$, $\mathrm{Pl}_m(\{x_i\}) = 0 \quad \forall x_i \notin A$, and $\mathrm{Pl}_m(\{x_i\}) = 1 \quad \forall x_i \in A$, then $\mathrm{TUM}(\mathcal{I}_m)$ must take the form:

$$\mathrm{TUM}(\mathcal{I}_m) = f(|A|),$$

$f : \mathbb{N} \to \mathbb{R}$ being an increasing function.

3. **Coherent Range**: $\mathrm{TUM}(\mathcal{I}_m)$ has to be non-negative.

It must hold that $\mathrm{TUM}(\mathcal{I}_m) = 0 \Leftrightarrow \mathrm{Bel}_m(\{x_i\}) = \mathrm{Pl}_m(\{x_i\}) = 1$ for some $i \in \{1, 2 \ldots, t\}$ and $\mathrm{Bel}_m\left(\{x_j\}\right) = \mathrm{Pl}_m\left(\{x_j\}\right) = 0 \; \forall j = 1, 2, \ldots, t, \quad j \neq i$.

The maximum value of $\mathrm{TUM}(\mathcal{I}_m)$ must be attained when $\mathrm{Bel}_m(\{x_i\}) = 0$ and $\mathrm{Pl}_m(\{x_i\}) = 1, \quad \forall i = 1, 2, \ldots, t$.

[3] These mathematical properties are adaptations of the properties established as crucial by Abellán and Masegosa [21] to uncertainty measures on belief intervals for singletons.

4. **Monotonicity**: Let m_1 and m_2 be two BPAs on X whose respective sets of belief intervals for singletons are $\mathcal{I}_{m_1}$ and $\mathcal{I}_{m_2}$. If it holds that:

$$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1, 2, \dots, t,$$

then TUM must verify that

$$\mathsf{TUM}(\mathcal{I}_{m_1}) \leqslant \mathsf{TUM}(\mathcal{I}_{m_2}).$$

5. **Subadditivity**: Let m be a BPA on a product space $X \times Y$ and $\mathcal{I}_m$ its associated set of belief intervals for singletons. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$ denote the projections of $\mathcal{I}_m$ on X and Y, respectively. Then, TUM must satisfy:

$$\mathsf{TUM}(\mathcal{I}_m) \leqslant \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow X}\right) + \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow Y}\right). \quad (8.5)$$

6. **Additivity**: Let m be a BPA on a product space $X \times Y$ and $\mathcal{I}_m$ its corresponding set of belief intervals for singletons. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$ be the projections of $\mathcal{I}_m$ on X and Y, respectively. Let $\mathcal{P}(\mathcal{I}_m)$, $\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$, and $\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$ denote the credal sets consistent with $\mathcal{I}_m$, $\mathcal{I}_m^{\downarrow X}$, and $\mathcal{I}_m^{\downarrow Y}$, respectively. If there is strong independence under $\mathcal{P}(\mathcal{I}_m)$, that is, $\mathcal{P}(\mathcal{I}_m) = \mathsf{CH}\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right)$, then TUM must verify the following equality:

$$\mathsf{TUM}(\mathcal{I}_m) = \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow X}\right) + \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow Y}\right).$$

As happens with total uncertainty measures on BPAs, in some cases, depending on the form of the uncertainty measure, it makes more sense to consider the submultiplicativity and multiplicativity properties than subadditivity and additivity. Such properties, for total uncertainty measures on belief intervals for singletons, are defined in the following way taking into account the definitions of additivity and subadditivity for this type of measures:

– **Submultiplicativity**: Let m be a BPA on a product space $X \times Y$ and $\mathcal{I}_m$ its associated set of belief intervals for singletons. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$ be the projections of $\mathcal{I}_m$ on X and Y, respectively. Then, TUM must verify that:

$$\mathsf{TUM}(\mathcal{I}_m) \leqslant \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathsf{TUM}\left(\mathcal{I}_m^{\downarrow Y}\right).$$

– **Multiplicativity**: Let m be a BPA on a product space $X \times Y$ and $\mathcal{I}_m$ its corresponding set of belief intervals for singletons. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$

be the projections of $\mathcal{I}_m$ on X and Y, respectively. Let $\mathcal{P}(\mathcal{I}_m)$, $\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$, and $\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$ denote the credal sets associated with $\mathcal{I}_m$, $\mathcal{I}_m^{\downarrow X}$, and $\mathcal{I}_m^{\downarrow Y}$, respectively. If there is strong independence under $\mathcal{P}(\mathcal{I}_m)$, i.e, $\mathcal{P}(\mathcal{I}_m) = \mathrm{CH}\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right)$, then TUM must satisfy the following equality:

$$\mathrm{TUM}(\mathcal{I}_m) = \mathrm{TUM}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathrm{TUM}\left(\mathcal{I}_m^{\downarrow Y}\right).$$

Concerning the behavioral requirements for total uncertainty measures on belief intervals for singletons, the following points must be considered:

- As pointed out before, belief intervals for singletons are easier to manage that BPAs to represent uncertainty-based information. Thereby, uncertainty measures on belief intervals for singletons are, generally, faster to compute than uncertainty measures on BPAs. Even so, a total uncertainty measure on belief intervals for singletons must not require a very complex calculation.

- When the belief intervals for singletons are utilized to quantify the uncertainty-based information, conflict and non-specificity also coexist, as stated by Wang and Song [212]. Consequently, a total uncertainty measure on belief intervals for singletons must not conceal both kinds of uncertainty, as happens with total uncertainty measures on BPAs.

  - On the one hand, according to Wang and Song [212], the non-specificity of a certain belief interval is measured via its span. But, indeed, it is wanted to measure the non-specificity of the whole set of belief intervals for singletons $\mathcal{I}_m$. The non-specificity value must be equal to 0 if, and only if, there is a unique probability distribution consistent with the belief intervals for singletons, that is, $\mathrm{Bel}_m(\{x_i\}) = \mathrm{Pl}_m(\{x_i\}) \quad \forall i = 1, 2, \ldots, t$. The maximum value of non-specificity must be attained when all probability distributions are compatible with the belief intervals for singletons, which happens if, and only if, $\mathrm{Bel}_m(\{x_i\}) = 0$ and $\mathrm{Pl}_m(\{x_i\}) = 1, \quad \forall i = 1, 2, \ldots, t$. Therefore, it makes sense that the non-specificity value of a total uncertainty measure on $\mathcal{I}_m$ indicates how large is the set of probability distributions compatible with $\mathcal{I}_m$.

  - On the other hand, the conflict of $\mathcal{I}_m$ indicates the distribution of the belief and plausibility values of the elements of X [212]. In consequence, the conflict value of a total uncertainty measure on belief intervals for singletons should be related with the Shannon entropy.

The maximum value of conflict must be obtained when $\mathcal{P}(\mathcal{I}_m)$ only contains the uniform probability distribution (in this case, there is no non-specificity). If the plausibility value for an element of X is equal to 1, then, due to the reachability of a set of belief intervals for singletons, the belief values for the rest of the elements of X are equal to 0. In these situations, a degenerate probability distribution is consistent with $\mathcal{I}_m$, and it can be considered that there is no conflict; the only type of uncertainty existing in these cases is non-specificity, which depends on how large is $\mathcal{P}(\mathcal{I}_m)$. Hence, it is logical that the conflict value of $\mathcal{I}_m$ coincides with the minimum conflict value between all the probability distributions compatible with $\mathcal{I}_m$.

- As happens with total uncertainty measures on BPAs, a total uncertainty measure on belief intervals for singletons must be sensitive to changes in such intervals. It must be remarked that, if a certain belief interval is widened (narrowed), then the non-specificity value may increase (decrease). In contrast, in these cases, the conflict value might decrease (increase). Thus, it makes sense that, when there are changes in the belief intervals for singletons, the total uncertainty value keeps equal and the conflict and non-specificity values vary. Hence, a total uncertainty measure on belief intervals for singletons has to be sensitive to changes in such intervals, directly or via its parts of conflict and non-specificity.

- Every total uncertainty measure on BPAs must be extensible to more general theories than ET. However, in most of such theories, the probabilistic knowledge can be expressed via a coherent lower probability function, which always has associated a coherent upper probability function. In consequence, in more general theories than ET, the lower and upper probability values for singletons can be considered. Thus, the extension of a total uncertainty measure on belief intervals for singletons to more general theories than ET is always possible.

In this way, every total uncertainty measure on $\mathcal{I}_m$, $TUM(\mathcal{I}_m)$, must satisfy the following behavioral requirements:

1. The calculation of $TUM(\mathcal{I}_m)$ must not be too complex.

2. It has to be possible to decompose $TUM(\mathcal{I}_m)$ into two measures that coherently indicate, respectively, the conflict and non-specificity values corresponding to $\mathcal{I}_m$.

3. $\mathsf{TUM}\left(\mathcal{I}_m\right)$ must be sensitive to changes in the belief intervals for singletons, directly or through its components of conflict and non-specificity.

8.3.1 Analysis of uncertainty measures on belief intervals for singletons

In this subsection, we analyze which of the total uncertainty measures on belief intervals for singletons proposed so far, described in Section 3.3.3, satisfy each one of the essential mathematical properties exposed above for total uncertainty measures on belief intervals for singletons.

- **Probabilistic consistency**: If $Bel_m\left(\{x_i\}\right) = Pl_m\left(\{x_i\}\right) \quad \forall i = 1, 2, \ldots, t$, it is easy to deduce that both $\mathsf{TUM}^I\left(\mathcal{I}_m\right)$ and $\mathsf{TUM}^I_E\left(\mathcal{I}_m\right)$ may differ from the Shannon entropy.

  In contrast, in these situations,
  $SU(\mathcal{I}_m) = -\sum_{i=1}^{t} Bel_m\left(\{x_i\}\right) \log_2\left(Bel_m\left(\{x_i\}\right)\right) \Rightarrow SU(\mathcal{I}_m)$ collapses to the Shannon entropy.

- **Generalized Set Consistency**: Suppose that $\exists A \subseteq X$ such that
  $Bel_m\left(\{x_i\}\right) = 0 \quad \forall i = 1, 2, \ldots, t, \quad Pl_m\left(\{x_i\}\right) = 1 \quad \forall x_i \in A$, and
  $Pl_m\left(\left\{x_j\right\}\right) = 0 \quad \forall x_j \notin A$.

  For $x_i \in A$, it holds that
  $d^I\left(\left[Bel_m\left(\{x_i\}\right), Pl_m\left(\{x_i\}\right)\right], [0, 1]\right) = d^I\left([0, 1], [0, 1]\right) = 0$. For $x_i \notin A$, it is satisfied that $d^I\left(\left[Bel_m\left(\{x_i\}\right), Pl_m\left(\{x_i\}\right)\right], [0, 1]\right) = d^I\left([0, 0], [0, 1]\right) = \frac{1}{\sqrt{3}}$. Hence, in these scenarios, $\mathsf{TUM}^I\left(\mathcal{I}_m\right) = 1 - \frac{1}{t}\sqrt{3}\sum_{x_i \notin A} \frac{1}{\sqrt{3}} = 1 - \frac{|\overline{A}|}{t} = \frac{|A|}{t} \Rightarrow \mathsf{TUM}^I$ satisfies Generalized Set Consistency.

  In these cases,

$$
\begin{aligned}
SU(\mathcal{I}_m) = \sum_{i=1}^{t} \Bigg[& -\frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2} \log_2\left(\frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2}\right) \\
& + \frac{Pl_m\left(\{x_i\}\right) - Bel_m\left(\{x_i\}\right)}{2}\Bigg] \\
= & \sum_{x_i \in A}\left[-\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\right] + \sum_{x_i \notin A} 0 = \sum_{x_i \in A} 1 = |A| \Rightarrow
\end{aligned}
$$

  SU verifies the Generalized Set Consistency property.

Concerning TUM_E^I, we have that, for $x_i \in A$, $d_E^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right) = d_E^I([0,1],[0,1]) = 0$. For $x_i \notin A$, $d_E^I\left([(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right) = d^I([0,0],[0,1]) = 1$. Therefore,

$$\begin{aligned}TUM_E^I(\mathcal{I}_m) &= \sum_{i=1}^{t}\left[1 - d_E^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right)\right] \\ &= \sum_{x_i \in A}\left[1 - d_E^I([0,1],[0,1])\right] + \sum_{x_i \notin A}\left[1 - d_E^I([0,0],[0,1])\right] \\ &= \sum_{x_i \in A} 1 - \sum_{x_i \notin A} 0 = |A|,\end{aligned}$$

which implies that TUM_E^I satisfies Generalized Set Consistency.

- **Coherent Range**: The range of TUM^I is equal to $[0,1]$ [223]. The minimum value of $d^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right)$ is reached when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1$. Such a minimum value is equal to 0. In consequence, when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1, \quad \forall i = 1,2,\ldots,t$, TUM^I attains its maximum value, which is equal to 1. The maximum value of $d^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right)$ is obtained when $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 0$ or $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$. In both cases, such a value is equal to $\sqrt{3}$. Thus, TUM^I is equal to 0 if, and only if, $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$ for some $i \in \{1,2,\ldots,t\}$ and $Bel_m\left(\{x_j\}\right) = Pl_m\left(\{x_j\}\right) = 0 \quad \forall j \in \{1,2,\ldots,t\} \setminus \{i\}$.

It can be checked that $d_E^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right) = 0 \Leftrightarrow Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1$. In this way, TUM_E^I obtains its maximum value, which is equal to t, when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1 \quad \forall i = 1,2,\ldots,t$. Now, $d_E^I\left([Bel_m(\{x_i\}), Pl_m(\{x_i\})], [0,1]\right) = 1$ if, and only if $\sqrt{(Bel_m(\{x_i\}))^2 + (1 - Pl_m(\{x_i\}))^2} = 1 \Leftrightarrow Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 0$ or $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$. So, TUM_E^I is equal to 0 if, and only if, $\exists i \in \{1,2,\ldots,t\}$ such that $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$ and $Bel_m\left(\{x_j\}\right) = Pl_m\left(\{x_j\}\right) = 0 \quad \forall j \in \{1,2,\ldots,t\} \setminus \{i\}$.

We may note that

$$\frac{Bel_m(\{x_i\}) + Pl_m(\{x_i\})}{2} \log_2\left(\frac{Bel_m(\{x_i\}) + Pl_m(\{x_i\})}{2}\right) = 0 \Leftrightarrow$$
$$Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 0 \vee Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1,$$
$$\forall i = 1,2,\ldots,t.$$

Therefore, SU is equal 0 if, and only if, $\exists i \in \{1,2,\ldots,t\}$ such that $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$ and $Bel_m\left(\{x_j\}\right) = Pl_m\left(\{x_j\}\right) = 0 \quad \forall j \in \{1,2,\ldots,t\} \setminus \{i\}$. The maximum value of $\frac{Pl_m(\{x_i\})-Bel_m(\{x_i\})}{2}$ is attained when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1 \quad \forall i = 1,2,\ldots,t$. In these situations, $-\frac{Bel_m(\{x_i\})+Pl_m(\{x_i\})}{2}\log_2\left(\frac{Bel_m(\{x_i\})+Pl_m(\{x_i\})}{2}\right)$ also reaches its maximum value, $\quad \forall i = 1,2,\ldots,t$, and, thus, the maximum value of SU is attained.

Consequently, the three total uncertainty measures on belief intervals for singletons proposed so far have a coherent range.

- **Monotonicity**: Let m_1 and m_2 be two BPAs on X and $\mathcal{I}_{m_1}$ and $\mathcal{I}_{m_2}$ their respective sets of belief intervals for singletons. Let us assume that

  $$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1,2,\ldots,t.$$

  Deng and Jiang [76] showed via counterexamples that, in these situations, it does not always hold that $SU(\mathcal{I}_{m_1}) \leqslant SU(\mathcal{I}_{m_2})$ nor that $TUM^I(\mathcal{I}_{m_1}) \leqslant TUM^I(\mathcal{I}_{m_2})$. In contrast, they demonstrated that $TUM^I_E(\mathcal{I}_{m_1}) \leqslant TUM^I_E(\mathcal{I}_{m_2})$ is always satisfied in these scenarios. Hence, TUM^I_E verifies the monotonicity property, unlike TUM^I and SU.

- **Subadditivity/submultiplivativity** and **additivity/multiplicativity**: Since $\log(a \times b) = \log(a) + \log(b) \quad \forall a, b \in \mathbb{R}$, for SU, the subadditivity and additivity properties make more sense than submultiplicativity and multiplicativity. In contrast, for the interval distance-based total uncertainty measures, the submultiplicativity and multiplicativity requirements are more appropriate than subadditivity and additivity [224].

  The following example shows that both TUM^I and TUM^I_E violate the submultiplicativity property.

Example 8.3.2 *Let* $X = \{x_1, x_2, x_2\}$ *and* $Y = \{y_1, y_2\}$ *be two finite sets. We denote* $z_{ij} = (x_i, y_j), \quad \forall i = 1,2,3, \quad j = 1,2$. *Let us consider the following BPA* m *on the product space* $X \times Y$*:*

$$m(z_{11}) = 0.8, \quad m(X \times Y) = 0.2.$$

It is disposed of the following set of belief intervals for singletons, $\mathcal{I}_m$*:*

$$z_{11} \to [0.8, 1]; \quad z_{12} \to [0, 0.2]; \quad z_{21} \to [0, 0.2];$$
$$z_{22} \to [0, 0.2]; \quad z_{31} \to [0, 0.2]; \quad z_{32} \to [0, 0.2].$$

The marginal set of belief intervals for singletons on X, $\mathcal{J}_m^{\downarrow X}$, *is given by:*

$$x_1 \to [0.8, 1]; \quad x_2 \to [0, 0.2]; \quad x_3 \to [0, 0.2].$$

The set of the projections of $\mathcal{J}_m$ *on* Y, $\mathcal{J}_m^{\downarrow Y}$, *is the following one:*

$$y_1 \to [0.8, 1]; \quad y_2 \to [0, 0.2].$$

We have:

$$\begin{aligned} \mathrm{TUM}^I(\mathcal{J}_m) &= 1 - \frac{\sqrt{3}}{6}\left(d^I([0.8,1],[0,1]) + 5 \times d^I([0,0.2],[0,1])\right) \\ &= 1 - \frac{\sqrt{3}}{6}\left(\frac{0.8}{\sqrt{3}} + 5 \times \frac{0.8}{\sqrt{3}}\right) = 0.2, \end{aligned}$$

$$\begin{aligned} \mathrm{TUM}^I\left(\mathcal{J}_m^{\downarrow X}\right) &= 1 - \frac{\sqrt{3}}{3}\left(d^I([0.8,1],[0,1]) + 2 \times d^I([0,0.4],[0,1])\right) \\ &= 1 - \frac{\sqrt{3}}{3} \times \left(\frac{0.8}{\sqrt{3}} + \frac{2 \times 0.6}{\sqrt{3}}\right) = \frac{1}{3}, \end{aligned}$$

$$\begin{aligned} \mathrm{TUM}^I\left(\mathcal{J}_m^{\downarrow Y}\right) &= 1 - \frac{\sqrt{3}}{2}\left(d^I([0.8,1],[0,1]) + d^I([0,0.6],[0,1])\right) \\ &= 1 - \frac{\sqrt{3}}{2} \times \left(\frac{0.8}{\sqrt{3}} + \frac{0.4}{\sqrt{3}}\right) = 0.4, \end{aligned}$$

$$\begin{aligned} \mathrm{TUM}_E^I(\mathcal{J}_m) &= \left(1 - d_E^I([0.8,1],[0,1])\right) + 5 \times \left(1 - d_E^I([0,0.2],[0,1])\right) \\ &= (1 - 0.8) + 5 \times (1 - 0.8) = 6 \times 0.2 = 1.2, \end{aligned}$$

$$\begin{aligned} \mathrm{TUM}_E^I\left(\mathcal{J}_m^{\downarrow X}\right) &= \left(1 - d_E^I([0.8,1],[0,1])\right) + 2 \times \left(1 - d_E^I([0,0.2],[0,1])\right) \\ &(1 - 0.8) + 2 \times (1 - 0.8) = 3 \times 0.2 = 0.6, \end{aligned}$$

$$\begin{aligned} \mathrm{TUM}_E^I\left(\mathcal{J}_m^{\downarrow Y}\right) &= \left(1 - d_E^I([0.8,1],[0,1])\right) + \left(1 - d_E^I([0,0.2],[0,1])\right) \\ &= 0.2 + 0.2 = 0.4. \end{aligned}$$

Hence,

$$\begin{aligned} \mathrm{TUM}^I\left(\mathcal{J}_m^{\downarrow X}\right) \times \mathrm{TUM}^I\left(\mathcal{J}_m^{\downarrow Y}\right) &= \frac{0.4}{3} < 0.2 = \mathrm{TUM}^I(\mathcal{J}_m), \\ \mathrm{TUM}_E^I\left(\mathcal{J}_m^{\downarrow X}\right) \times \mathrm{TUM}_E^I\left(\mathcal{J}_m^{\downarrow Y}\right) &= 0.24 < 1.2 = \mathrm{TUM}_E^I(\mathcal{J}_m). \end{aligned}$$

In the following example, it is shown that TUM^I and TUM^I_E do neither satisfy the multiplicativity property:

Example 8.3.3 *Suppose that* $X = \{x_1, x_2, x_2\}$ *and* $Y = \{y_1, y_2\}$ *are two finite sets and that we have the following BPA* m *on* $X \times Y$*:*

$$m(\{z_{11}\}) = \frac{1}{3}, \quad m(\{z_{21}\}) = \frac{1}{3}, \quad m(\{z_{31}\}) = \frac{1}{3},$$

where $z_{ij} = (x_i, y_j), \quad \forall i = 1, 2, 3, \quad j = 1, 2.$

The set of belief intervals for singletons, $\mathcal{I}_m$*, is given by:*

$$z_{11} \to \left[\frac{1}{3}, \frac{1}{3}\right]; \quad z_{12} \to [0, 0]; \quad z_{21} \to \left[\frac{1}{3}, \frac{1}{3}\right],$$
$$z_{22} \to [0, 0]; \quad z_{31} \to \left[\frac{1}{3}, \frac{1}{3}\right]; \quad z_{32} \to [0, 0].$$

Let $\mathcal{I}_m^{\downarrow X}$ *and* $\mathcal{I}_m^{\downarrow Y}$ *denote the projections of* $\mathcal{I}_m$ *on* X *and* Y*, respectively. They are determined by:*

$$x_1 \to \left[\frac{1}{3}, \frac{1}{3}\right]; \quad x_2 \to \left[\frac{1}{3}, \frac{1}{3}\right]; \quad x_3 \to \left[\frac{1}{3}, \frac{1}{3}\right];$$

$$y_1 \to [1, 1]; \quad y_2 \to [0, 0].$$

It is easy to observe that, in this case, $\mathcal{P}(\mathcal{I}_m) = \mathsf{CH}\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right)$*. It holds that:*

$$\begin{aligned}\mathsf{TUM}^I(\mathcal{I}_m) &= 1 - \frac{\sqrt{3}}{6}\left(3 \times d^I\left(\left[\frac{1}{3}, \frac{1}{3}\right], [0, 1]\right) + 3 \times d^I([0, 0], [0, 1])\right)\\ &= 1 - \frac{\sqrt{3}}{6} \times \left(3 \times \sqrt{\left(\frac{1}{6}\right)^2 + \frac{(0.5)^2}{3}} + \frac{3}{\sqrt{3}}\right) = 0.3354,\end{aligned}$$

$$\begin{aligned}\mathsf{TUM}^I\left(\mathcal{I}_m^{\downarrow X}\right) &= 1 - \frac{\sqrt{3}}{3}\left(3 \times d^I\left(\left[\frac{1}{3}, \frac{1}{3}\right], [0, 1]\right)\right)\\ &= 1 - \sqrt{3} \times \left(\sqrt{\left(\frac{1}{6}\right)^2 + \frac{(0.5)^2}{3}}\right) = 0.6709,\end{aligned}$$

$$\begin{aligned}\mathrm{TUM}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow Y}\right) &= 1-\frac{\sqrt{3}}{2}\left(d^{\mathrm{I}}\left([1,1],[0,1]\right)+d^{\mathrm{I}}\left([0,0],[0,1]\right)\right)\\ &= 1-\frac{\sqrt{3}}{2}\times\frac{2}{\sqrt{3}}=0.\end{aligned}$$

$$\begin{aligned}\mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m\right) = 3\times&\left(1-d_{\mathrm{E}}^{\mathrm{I}}\left(\left[\frac{1}{3},\frac{1}{3}\right],[0,1]\right)\right)\\ &+3\times\left(1-d_{\mathrm{E}}^{\mathrm{I}}\left([0,0],[0,1]\right)\right)=0.7639,\end{aligned}$$

$$\mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow X}\right)=3\times\left(1-d_{\mathrm{E}}^{\mathrm{I}}\left(\left[\frac{1}{3},\frac{1}{3}\right],[0,1]\right)\right)=0.7639,$$

$$\mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow Y}\right)=1-d_{\mathrm{E}}^{\mathrm{I}}\left([1,1],[0,1]\right)+1-d_{\mathrm{E}}^{\mathrm{I}}\left([0,0],[0,1]\right)=0.$$

Thereby,

$$\begin{aligned}\mathrm{TUM}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow X}\right)\times\mathrm{TUM}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow Y}\right)&=0.6709\times 0=0\neq 0.3354=\mathrm{TUM}^{\mathrm{I}}\left(\mathcal{I}_m\right),\\ \mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow X}\right)\times\mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m^{\downarrow Y}\right)&=0.7639\times 0=0\neq 0.7639=\mathrm{TUM}_{\mathrm{E}}^{\mathrm{I}}\left(\mathcal{I}_m\right).\end{aligned}$$

The following example shows that SU does not satisfy the subadditivity requirement.

Example 8.3.4 *Let* $X=\{x_1,x_2,x_3\}$ *and* $Y=\{y_1,y_2\}$ *be two finite sets and* m *the following BPA on the product space* $X\times Y$*:*

$$m\left(z_{11},z_{12},z_{21},z_{22}\right)=0.7,\quad m\left(z_{31},z_{32}\right)=0.1,\quad m(X\times Y)=0.2,$$

where $z_{ij}=(x_i,y_j),\quad\forall i=1,2,3,\quad j=1,2.$

It is disposed of the following set of belief intervals for singletons, $\mathcal{I}_m$*:*

$$\begin{aligned}&z_{11}\to[0,0.9];\quad z_{12}\to[0,0.9];\quad z_{21}\to[0,0.9];\\ &z_{22}\to[0,0.9];\quad z_{31}\to[0,0.3];\quad z_{32}\to[0,0.3].\end{aligned}$$

We consider the projections of the belief intervals for singletons on X *and* Y*, denoted by* $\mathcal{I}_m^{\downarrow X}$ *and* $\mathcal{I}_m^{\downarrow Y}$*, respectively:*

$$x_1\to[0,1];\quad x_2\to[0,1];\quad x_3\to[0,0.6];$$

$$y_1 \to [0,1]; \quad y_2 \to [0,1].$$

It holds that:

$$\begin{aligned}
&SU(\mathcal{I}_m) = 4 \times \left(-\frac{0.9}{2}\log_2\left(\frac{0.9}{2}\right) + \frac{0.9}{2}\right)\\
&+ 2 \times \left(-\frac{0.3}{2}\log_2\left(\frac{0.3}{2}\right) + \frac{0.3}{2}\right) = 4.9947,\\
&SU\left(\mathcal{I}_m^{\downarrow X}\right) = 2 \times \left(-\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\right) - 0.3\log_2 0.3 = 2.5211,\\
&SU\left(\mathcal{I}_m^{\downarrow Y}\right) = 2 \times \left(-\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\right) = 2.
\end{aligned}$$

In this way,

$$SU(\mathcal{I}_m) = 4.9947 > 4.5211 = 2.5211 + 2 = SU\left(\mathcal{I}_m^{\downarrow X}\right) + SU\left(\mathcal{I}_m^{\downarrow Y}\right).$$

We show with an example below that SU does also not verify additivity.

Example 8.3.5 *Let* $X = \{x_1, x_2, x_3\}$ *and* $Y = \{y_1, y_2\}$ *be two finite sets. Suppose that we have the following BPA* m *on the product space* $X \times Y$*:*

$$m(X \times Y) = 1.$$

We shall denote $z_{ij} = (x_i, y_j) \quad \forall i = 1,2,3, \quad j = 1,2$*. The set of belief intervals for singletons,* $\mathcal{I}_m$*, is given by:*

$$\begin{aligned}
&z_{11} \to [0,1]; \quad z_{21} \to [0,1]; \quad z_{12} \to [0,1];\\
&z_{22} \to [0,1]; \quad z_{31} \to [0,1]; \quad z_{32} \to [0,1].
\end{aligned}$$

The projections of $\mathcal{I}_m$ *on* X *and* Y*, denoted by* $\mathcal{I}_m^{\downarrow X}$ *and* $\mathcal{I}_m^{\downarrow Y}$*, respectively, are the following ones:*

$$x_1 \to [0,1]; \quad x_2 \to [0,1] \quad x_3 \to [0,1];$$

$$y_1 \to [0,1]; \quad y_2 \to [0,1].$$

It is obvious that, in this case, $\mathcal{P}(\mathcal{I}_m) = CH\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right)$*. We have that:*

$$SU(\mathcal{I}_m) = 6 \times \left(-\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\right) = 6 \times 1 = 6,$$

$$\mathrm{SU}\left(\mathcal{I}_m^{\downarrow X}\right) = 3 \times \left(-\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\right) = 3 \times 1 = 3,$$

$$\mathrm{SU}\left(\mathcal{I}_m^{\downarrow Y}\right) = 2 \times \left(-\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\right) = 2 \times 1 = 2.$$

Therefore,

$$\mathrm{SU}\left(\mathcal{I}_m\right) = 6 \neq 5 = \mathrm{SU}\left(\mathcal{I}_m^{\downarrow X}\right) + \mathrm{SU}\left(\mathcal{I}_m^{\downarrow Y}\right).$$

With regard to the behavioral requirements, we must remark the following issues:

- Once it is disposed of the belief intervals for singletons, the computation of TUM^I, TUM_E^I, and SU is direct.
- So far, it has not been possible to decompose the total uncertainty measures that employ distance functions of belief intervals, TUM^I and TUM_E^I, into two measures that respectively quantify conflict and non-specificity.

  In contrast, SU can be rewritten as follows:

  $$\mathrm{SU}\left(\mathcal{I}_m\right) = \sum_{i=1}^{t} -\frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2}\log_2\left(\frac{Bel_m\left(\{x_i\}\right) + Pl_m\left(\{x_i\}\right)}{2}\right) + \sum_{i=1}^{t}\frac{Pl_m\left(\{x_i\}\right) - Bel_m\left(\{x_i\}\right)}{2}.$$

  The first term of the previous expression indicates conflict, whereas the second one captures non-specificity. In fact, the second term is equal to 0 if, and only if, $Bel_m\left(\{x_i\}\right) = Pl_m\left(\{x_i\}\right) \quad \forall i = 1, 2, \ldots, t$ (the span of all belief intervals for singletons is equal to 0). Also, the first term indicates how the belief and plausibility values for singletons are distributed. However, when $\exists i \in \{1, 2, \ldots, t\}$ such that $Pl_m\left(\{x_i\}\right) = 1$, the conflict value indicated by SU might not be equal to 0. It is undesirable because, in these cases, there is no conflict in the belief intervals for singletons, as argued previously.
- It is easy to observe that the distance functions utilized in TUM^I and TUM_E^I are sensitive to variations in the belief and plausibility values for singletons. In consequence, both uncertainty measures are directly sensitive to changes in the belief intervals for singletons.

Also, the values $-\frac{Bel_m(\{x_i\})+Pl_m(\{x_i\})}{2}\log_2\left(\frac{Bel_m(\{x_i\})+Pl_m(\{x_i\})}{2}\right)$ and $\frac{Pl_m(\{x_i\})-Bel_m(\{x_i\})}{2}$ may vary when the belief and plausibility values for singletons change, $\forall i = 1, 2 \ldots, t$. Thus, SU is sensitive to changes in the belief intervals for singletons via its parts of conflict and non-specificity.

Table 8.1 summarizes the mathematical properties satisfied by the total uncertainty measures on belief intervals developed so far. Likewise, Table 8.2 shows a summary about the behavioral requirements of such measures.

Table 8.1: Summary of the mathematical properties satisfied by the total uncertainty measures on belief intervals for singletons proposed so far.

Property	TUM^I	TUM^I_E	SU
Probabilistic Consistency	No	No	Yes
Generalized Set Consistency	Yes	Yes	Yes
Coherent Range	Yes	Yes	Yes
Monotonicity	No	Yes	No
Subadditivity/Submultiplicativity	No	No	No
Additivity/Multiplicativity	No	No	No

Table 8.2: Summary of the behavioral requirements of the total uncertainty measures on belief intervals for singletons proposed so far.

Behavioral requirement	TUM^I	TUM^I_E	SU
Complexity	Low	Low	Low
Separation	No	No	Improvable
Sensitivity	Yes	Yes	Yes

We must remark the following issues about the mathematical properties:

- The three total uncertainty measures on belief intervals for singletons provide a logical result when it is only known that the information is focused on a subset of the set of possible alternatives since they satisfy Generalized Set Consistency.

- The ranges of TUM^I, TUM^I_E, and SU are all coherent.

- When the belief intervals for singletons are reduced to a single probability distribution, SU obtains a logical result, which coincides with the well-established uncertainty value for probability distributions (the Shannon entropy). It does not occur with the total uncertainty measures based on distance functions of intervals.

- Unlike TUM_E^I, TUM^I and SU are not always consistent with an increase or decrease of uncertainty-based information expressed via the belief intervals for singletons (Monotonicity property).
- All total uncertainty measures on belief intervals for singletons violate subadditivity and additivity, which implies that they might not produce coherent results when they are defined over a set of belief intervals for singletons on a product space that can be decomposed into more simple sets.
- Consequently, none of TUM^I, TUM_E^I, and SU satisfies all the required mathematical properties for total uncertainty measures on belief intervals for singletons.

Regarding the behavioral requirements, the following points should be noted:

- The computations of TUM^I, TUM_E^I, and SU are direct.
- SU can be decomposed into two measures that respectively capture conflict and non-specificity, unlike the intervals distance-based total uncertainty measures. Nevertheless, such a decomposition is not very coherent as the conflict value of SU might not be equal to 0 when the plausibility value for a singleton is equal to 1, which is not very logical.
- The three total uncertainty measures on belief intervals for singletons proposed so far are sensitive to changes in the belief intervals for singletons, directly or through the parts of conflict and non-specificity.

Therefore, none of the total uncertainty on belief intervals for singletons proposed so far satisfies all the essential mathematical properties and behavioral requirements for this kind of measure.

8.4 Maximum entropy on belief intervals for singletons

Let m be a BPA on X and Bel_m and Pl_m its associated belief and plausibility functions, respectively. Let $\mathcal{I}_m$ denote the set of belief intervals for singletons corresponding to m, determined by means of Equation (2.41). Let $\mathcal{P}(\mathcal{I}_m)$ be the credal set associated with $\mathcal{I}_m$, given by Equation (2.42).

We propose a total uncertainty measure that consists of the maximum entropy on the credal set compatible with the belief intervals for singletons:

$$S^{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)=\max_{p\in\mathcal{P}\left(\mathcal{I}_{m}\right)}\{S(p)\}, \tag{8.6}$$

$S(p)$ being the Shannon entropy of the probability distribution p.

This measure can be disaggregated as follows:

$$S^{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)=S_{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)+\left(S^{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)-S_{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)\right), \tag{8.7}$$

where $S_{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)$ is the minimum entropy on $\mathcal{P}\left(\mathcal{I}_{m}\right)$:

$$S_{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)=\min_{p\in\mathcal{P}\left(\mathcal{I}_{m}\right)}\{S(p)\}. \tag{8.8}$$

The first term of Equation (8.7) captures conflict whereas the second one indicates non-specificity.

For the computation of $S^{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)$, we utilize the algorithm proposed so far for the maximum entropy on a reachable set of probability intervals (Algorithm 3. Recall that the set of belief intervals for singletons is always reachable). Hence, Algorithm 12 details the procedure to obtain the probability distribution for which $S^{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)$ is attained. In it, *sec_min* indicates the second minimum value (-1 if such a second minimum value does not exist).

In order to compute $S_{*}\left(\mathcal{P}\left(\mathcal{I}_{m}\right)\right)$, we use the following lemma, proved by Wasserman and Kadane [213]:

Lemma 8.4.1 *Let* p *and* q *be two probability distributions on* X. *We denote* $p_i = p\left(\{x_i\}\right)$ *and* $q_i = q\left(\{x_i\}\right) \quad \forall i = 1,2,\dots,t$, *in such a way that* $p = (p_1,p_2,\dots,p_t)$ *and* $q = (q_1,q_2,\dots,q_t)$. *Let* p^* *(*q^**) be the array* p *(*q*) ordered decreasingly. If* $\sum_{i=1}^{j} p_i^* \leqslant \sum_{i=1}^{j} q_i^* \quad \forall j = 1,2\dots,t$, *then* $S(p) \geqslant S(q)$.

Let $\overline{p}_{\mathcal{I}_m}$ denote the probability distribution of minimum entropy on $\mathcal{P}\left(\mathcal{I}_{m}\right)$ and $\left((Bel_m)_1, (Bel_m)_2, \dots, (Bel_m)_t\right)$ and $\left((Pl_m)_1, (Pl_m)_2, \dots, (Pl_m)_t\right)$ the array of belief and plausibility values for singletons, respectively, where $(Bel_m) = Bel_m\left(\{x_i\}\right)$ and $(Pl_m) = Pl_m\left(\{x_i\}\right), \quad \forall i = 1,2,\dots,t$. Let Bel_m^* (Pl_m^*) be the array of belief (plausibility) values for singletons ordered decreasingly. Let Bel'_m denote the array of belief values for singletons ordered in the same way as Pl_m^*. Let $\overline{p}^*_{\mathcal{I}_m}$ be the array of $\overline{p}_{\mathcal{I}_m}$ ordered decreasingly. Then, $\overline{p}_{\mathcal{I}_m}$ can be obtained via Algorithm 13.

We may note that, at the end of the while loop of the algorithm, $r \leqslant t$ because $\mathcal{I}_m$ is reachable. The following result demonstrates that the probability distribution obtained in Algorithm 13 attains the minimum entropy on $\mathcal{P}\left(\mathcal{I}_{m}\right)$.

Algorithm 12: Procedure to compute the probability distribution of maximum entropy on the credal set corresponding to the set of belief intervals for singletons.

```
Procedure Determine probability distribution of maximum entropy on a set of belief intervals for singletons(Set of belief intervals for singletons $\mathcal{I}_m = \{[Bel_m(\{x_i\}), Pl_m(\{x_i\})], \quad i = 1, 2, \dots, t\}$)
for $i = 1$ to $t$ do
    $\hat{p}_{\mathcal{I}_m}(x_i) \leftarrow Bel_m(\{x_i\})$
$sum \leftarrow \sum_{i=1}^{t} \hat{p}_{\mathcal{I}_m}(x_i)$
while $sum < 1$ do
    $min\_prob \leftarrow \min_{i \in \{1,2,\dots,t\} | \hat{p}_{\mathcal{I}_m}(x_i) < Pl_m(\{x_i\})} \hat{p}_{\mathcal{I}_m}(x_i)$
     $index\_min\_prob \leftarrow \{i \in \{1, 2, \dots, t\} \mid \hat{p}_{\mathcal{I}_m}(x_i) = min\_prob\}$
     $num\_min \leftarrow |index\_min\_prob|$
     $sec\_min\_prob \leftarrow sec\_min_{i \in \{1,2,\dots,t\} | \hat{p}_{\mathcal{I}_m}(x_i) < u_i} \hat{p}_{\mathcal{I}_m}(x_i)$
    for $i \in index\_min\_prob$ do
        if $sec\_min\_prob = -1$ then
            $\hat{p}_{\mathcal{I}_m}(x_i) \leftarrow \hat{p}_{\mathcal{I}_m}(x_i) + \min\left(Pl_m(\{x_i\}) - \hat{p}_{\mathcal{I}_m}(x_i), \frac{1-sum}{num\_min}, 1\right)$
        else
            $\hat{p}_{\mathcal{I}_m}(x_i) \leftarrow \hat{p}_{\mathcal{I}_m}(x_i) +$
              $\min\left(Pl_m(\{x_i\}) - \hat{p}_{\mathcal{I}_m}(x_i), sec\_min\_prob - min\_prob, \frac{1-sum}{num\_min}\right)$
    $sum \leftarrow \sum_{i=1}^{t} \hat{p}_{\mathcal{I}_m}(x_i)$
return $\hat{p}_{\mathcal{I}_m}$
```

Algorithm 13: Procedure to compute the probability distribution of minimum entropy on the set of belief intervals for singletons.

```
Procedure Determine probability distribution of minimum entropy on a set of belief intervals for singletons(Set of belief intervals for singletons $\mathcal{I}_m = \{[Bel_m(\{x_i\}), Pl_m(\{x_i\})], \quad i = 1, 2, \ldots, t\}$)
for $i = 1$ to $t$ do
    $\overline{p}_{\mathcal{I}_m}(x_i) \leftarrow Bel_m(\{x_i\})$
$mass \leftarrow 1 - \sum_{i=1}^{t} Bel_m(\{x_i\})$
$r \leftarrow 1$
first_step ← false
while first_step = false do
    if $(Pl^*_m)_r - (Bel'_m)_r < mass$ then
        $(\overline{p}^*_{\mathcal{I}_m})_r \leftarrow (Pl^*_m)_r$
        $mass \leftarrow mass - (Pl^*_m)_r + (Bel'_m)_r$
        $r \leftarrow r + 1$
    else
        first_step ← true
    $k \leftarrow \arg\max_{o \geqslant r}\{(Bel'_m)_o + mass\}$
    $(\overline{p}^*_{\mathcal{I}_m})_r \leftarrow (Bel'_m)_k + mass$
return $\overline{p}_{\mathcal{I}_m}$
```

Theorem 8.4.1 *The probability distribution $\overline{p}_{\mathcal{I}_m}$, obtained in Algorithm 13, satisfies* $S(\overline{p}_{\mathcal{I}_m}) = S_*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$

Proof: With same notation as in Algorithm 13, it holds that:

$$\overline{p}^*_{\mathcal{I}_m} = \left((Pl^*_m)_1, \ldots, (Pl^*_m)_{r-1}, \alpha_r, \ \left(Bel''_m\right)_{r+1}, \ldots, \left(Bel''_m\right)_t\right),$$

where

$$\left(Bel''_m\right)_i \in \{(Bel_m)_1, (Bel_m)_2, \ldots, (Bel_m)_t\} \quad \forall i = r+1, \ldots, t,$$

$\left(Bel''_m\right)_i \geqslant \left(Bel''_m\right)_j \quad \forall i,j \in \{r+1, \ldots, t\}$ with $j \leqslant i$, and $\alpha_r \in [(Bel'_m)_r, (Pl^*_m)_r]$.

Suppose now that $q \in \mathcal{P}\left(\mathcal{I}_m\right)$ and let q^* be its corresponding array ordered decreasingly.

For $j = 1, 2, \ldots, r-1$ clearly:

$$\sum_{i=1}^{j} \left(\overline{p}^*_{\mathcal{I}_m}\right)_i = \sum_{i=1}^{j} (Pl^*_m)_i \geqslant \sum_{i=1}^{j} q^*_i.$$

For $j = r, r+1, \ldots, t$:

$$\sum_{i=1}^{j} \left(\overline{p}^*_{\mathcal{I}_m}\right)_i = 1 - \sum_{i=j+1}^{t} \left(Bel''_m\right)_i \geqslant 1 - \sum_{i=j+1}^{t} q^*_i = \sum_{i=1}^{j} q^*_i.$$

In consequence,

$$\sum_{i=1}^{j} \left(\overline{p}^*_{\mathcal{I}_m}\right)_i \geqslant \sum_{i=1}^{j} q^*_i, \quad \forall j = 1, 2, \ldots, t.$$

Due to Lemma 8.4.1, $S\left(\overline{p}_{\mathcal{I}_m}\right) \leqslant S(q)$. Thus, we can conclude that $\overline{p}_{\mathcal{I}_m}$ is the probability distribution of minimum entropy on $\mathcal{P}\left(\mathcal{I}_m\right)$. □

We show below an example about the procedure to obtain $S_*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$.

Example 8.4.1 *Let* $X = \{x_1, x_2, x_3, x_4\}$ *be the finite set. Let us consider the following BPA* m *on X:*

$$m\left(\{x_1\}\right) = 0.1, \quad m\left(\{x_2, x_3\}\right) = 0.6,$$

$$m\left(\{x_1, x_4\}\right) = 0.3.$$

We have the following set of belief intervals for singletons, $\mathcal{I}_m$*:*

$$x_1 \to [0.1, 0.4]; \quad x_2 \to [0, 0.6]; \quad x_3 \to [0, 0.6]; \quad x_4 \to [0, 0.3].$$

If we carry out the steps of the previous algorithm, we obtain the following values for $\overline{p}_{\mathcal{I}_m} = \left(\overline{p}_{\mathcal{I}_m}(x_1), \overline{p}_{\mathcal{I}_m}(x_2), \overline{p}_{\mathcal{I}_m}(x_3), \overline{p}_{\mathcal{I}_m}(x_4)\right)$*, the probability distribution of minimum entropy on the credal set corresponding to these intervals:*
$\overline{p}_{\mathcal{I}_m} = (0.1, 0, 0, 0)$,
$\overline{p}_{\mathcal{I}_m} = (0.1, 0, 0.6, 0)$,
$\overline{p}_{\mathcal{I}_m} = (0.4, 0, 0.6, 0)$.

8.4.1 Mathematical properties of our proposal

In this subsection, we analyze which of the crucial mathematical properties for total uncertainty measures on belief intervals for singletons, exposed in Section 8.3, are satisfied by our proposed measure $S^*(\mathcal{P}(\mathcal{I}_m))$.

- **Probabilistic consistency**: If $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) \quad \forall i = 1, 2, \dots, t$, then $\mathcal{P}(I_m)$ contains a unique probability distribution, given by $p(x_i) = Bel_m(\{x_i\}), \quad \forall i = 1, 2, \dots, t$. Clearly, in these cases, $S^*(\mathcal{P}(I_m))$ coincides with the Shannon entropy.

  Generalized Set Consistency: Suppose that $\exists A \subseteq X$ with $|A| \geqslant 2$ such that $Bel_m(\{x_i\}) = 0 \quad \forall i = 1, 2, \dots, t, \quad Pl_m(\{x_i\}) = 1 \quad \forall x_i \in A$, and $Pl_m\left(\{x_j\}\right) = 0 \quad \forall x_j \notin A$. We may observe that, in this situation, the probability distribution of maximum entropy, among the ones belonging to $\mathcal{P}(\mathcal{I}_m)$, is given by:

$$\hat{p}_{\mathcal{I}_m}(x_i) = \begin{cases} \frac{1}{|A|} & \text{if} \quad x_i \in A \\ \\ 0 & \text{if} \quad x_i \notin A \end{cases}$$

  It holds that

$$\begin{aligned} S^*(\mathcal{P}(\mathcal{I}_m)) = S(\hat{p}_{\mathcal{I}_m}) &= -\sum_{x_i \in A} \frac{1}{|A|} \log_2\left(\frac{1}{|A|}\right) \\ &= |A| \frac{1}{|A|} \log_2(|A|) = \log_2(|A|). \end{aligned}$$

  As log_2 is an increasing function, it is deduced that $S^*(\mathcal{P}(\mathcal{I}_m))$ satisfies Generalized Set Consistency.

- **Coherent range**: The minimum value of $S^*(\mathcal{P}(\mathcal{I}_m))$ is equal to 0. It is obtained if, and only if, $\mathcal{P}(\mathcal{I}_m)$ only contains a degenerate probability distribution. It is easy to deduce that it happens if, and only if, $Bel_m(\{x_i\}) = Pl_m(\{x_i\}) = 1$ for some $i \in \{1, 2, \dots, t\}$ and $Bel_m\left(\{x_j\}\right) = Pl_m\left(\{x_j\}\right) =$

$0 \quad \forall j \in \{1, 2, \ldots, t\}, \quad j \neq i$. Furthermore, when all the probability distributions on X belong to $\mathcal{P}(\mathcal{I}_m)$, that is, when $Bel_m(\{x_i\}) = 0$ and $Pl_m(\{x_i\}) = 1 \quad \forall i = 1, 2, \ldots, t$, $S^*(\mathcal{P}(\mathcal{I}_m))$ attains its maximum value ($\log_2(|X|)$). In consequence, the range of $S^*(\mathcal{P}(\mathcal{I}_m))$ is coherent.

- **Monotonicity**: Let m_1 and m_2 be two BPAs on X and $\mathcal{I}_{m_1}$ and $\mathcal{I}_{m_2}$ their respective sets of belief intervals for singletons. Let us assume that

$$[Bel_{m_1}(\{x_i\}), Pl_{m_1}(\{x_i\})] \subseteq [Bel_{m_2}(\{x_i\}), Pl_{m_2}(\{x_i\})], \quad \forall i = 1, 2, \ldots, t.$$

From Proposition 8.3.1, it follows that $\mathcal{P}(\mathcal{I}_{m_1}) \subseteq \mathcal{P}(\mathcal{I}_{m_2})$ and, obviously, $S^*(\mathcal{P}(\mathcal{I}_{m_1})) \leqslant S^*(\mathcal{P}(\mathcal{I}_{m_2}))$.

- **Subadditivity and additivity**: Let $X = \{x_1, x_2, \ldots, x_t\}$ and $Y = \{y_1, y_2, \ldots, y_{t'}\}$ be two finite sets and m a BPA on the product space $X \times Y$. Let $\mathcal{I}_m = \left\{\left[l_{ij}^m, u_{ij}^m\right], \quad i = 1, 2, \ldots, t, \quad j = 1, 2, \ldots, t'\right\}$ be the set of belief intervals for singletons associated with m and $\mathcal{P}(\mathcal{I}_m)$ the corresponding credal set. Let $\mathcal{I}_m^{\downarrow X}$ and $\mathcal{I}_m^{\downarrow Y}$ denote the marginal sets of intervals of $\mathcal{I}_m$ on X and Y, respectively, determined through Proposition 2.2.9. Let $\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$ and $\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$ denote the credal sets consistent with such sets of intervals and $\mathcal{P}^{\downarrow X}(\mathcal{I}_m)$ and $\mathcal{P}^{\downarrow Y}(\mathcal{I}_m)$ the marginal credal sets of $\mathcal{P}(\mathcal{I}_m)$ on X and Y, respectively.

In the following proposition, we demonstrate that projecting on the belief intervals for singletons is equivalent to projecting on the corresponding credal set:

Proposition 8.4.1 *It is satisfied that*
$\mathcal{P}^{\downarrow X}(\mathcal{I}_m) = \mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right), \quad \mathcal{P}^{\downarrow Y}(\mathcal{I}_m) = \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$

Proof: Let $p_X \in \mathcal{P}^{\downarrow X}(\mathcal{I}_m)$. Then, $\exists p \in \mathcal{P}(\mathcal{I}_m)$ such that $p_X(x_i) = \sum_{j=1}^{t'} p\left(x_i, y_j\right) \quad \forall i = 1, 2, \ldots, t$.

Since $p \in \mathcal{P}(\mathcal{I}_m)$, we have that

$$l_{ij}^m \leqslant p\left(x_i, y_j\right) \leqslant u_{ij}^m, \quad \forall i = 1, 2, \ldots, t, \quad j = 1, \ldots, t' \Rightarrow$$
$$\sum_{j=1}^{t'} l_{ij}^m \leqslant \sum_{j=1}^{t'} p\left(x_i, y_j\right) = p_X(x_i) \leqslant \sum_{j=1}^{t'} u_{ij}^m \quad \forall i = 1, 2, \ldots, t,$$

which implies that $p_X \in \mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$.

Suppose now that $p_X \in \mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)$. Then:

$$\sum_{j=1}^{t'} l_{ij}^m \leqslant p_X\left(x_i\right) \leqslant \sum_{j=1}^{t'} u_{ij}^m, \quad \forall i = 1, 2, \ldots, t.$$

For each $i = 1, 2, \ldots, t$, there are 3 possibilities:

1. $p_X\left(x_i\right) = \sum_{j=1}^{t'} l_{ij}^m$
2. $p_X\left(x_i\right) = \sum_{j=1}^{t'} u_{ij}^m$
3. $p_X\left(x_i\right) = \lambda_i$, where $\sum_{j=1}^{t'} l_{ij}^m < \lambda_i < \sum_{j=1}^{t'} u_{ij}^m$.

   We consider:

$$p\left(x_i, y_j\right) = \left\{ \begin{array}{lll} l_{ij}^m & \text{if} & p_X\left(x_i\right) = \sum_{j=1}^{t'} l_{ij}^m, \\ u_{ij}^m & \text{if} & p_X\left(x_i\right) = \sum_{j=1}^{t'} u_{ij}^m, \\ \alpha_{ij} & \text{if} & p_X\left(x_i\right) = \lambda_i \end{array} \right\}$$

with $\sum_{j=1}^{t'} l_{ij}^m < \lambda_i < \sum_{j=1}^{t'} u_{ij}^m$, $l_{ij}^m \leqslant \alpha_{ij} \leqslant u_{ij}^m$, in such a way that $\sum_{j=1}^{t'} \alpha_{ij} = \lambda_i \quad \forall i \in \{1, 2, \ldots, t\}$ such that $\sum_{j=1}^{t'} l_{ij}^m < p_X\left(x_i\right) < \sum_{j=1}^{t'} u_{ij}^m$.

Clearly, $p \in \mathcal{P}\left(\mathcal{I}_m\right)$ and $p_X\left(x_i\right) = \sum_{j=1}^{t'} p\left(x_i, y_j\right) \quad \forall i = 1, 2, \ldots, t$. Consequently, $p_X \in \mathcal{P}^{\downarrow X}\left(\mathcal{I}_m\right)$.

The proof of $\mathcal{P}^{\downarrow Y}\left(\mathcal{I}_m\right) = \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)$ is analogous.

□

The following proposition shows that $S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$ verifies the subadditivity property:

Proposition 8.4.2 *With the above notation, it always holds that*

$$S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) \leqslant S^*\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)\right) + S^*\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right).$$

Taking Proposition 8.4.1 into account, the proof of this result is identical to the one given by Abellán and Moral [1] for the subadditivity property for the maximum entropy on general credal sets.

$S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$ also satisfies additivity, as shown in the following result.

Proposition 8.4.3 *With the above notation, if there is strong independence under* $\mathcal{P}(\mathcal{I}_m)$*, that is* $\mathcal{P}(\mathcal{I}_m) = \mathrm{CH}\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right) \times \mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right)$*, then*

$$S^*\left(\mathcal{P}(\mathcal{I}_m)\right) = S^*\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow X}\right)\right) + S^*\left(\mathcal{P}\left(\mathcal{I}_m^{\downarrow Y}\right)\right).$$

The proof of this proposition is identical to the one provided by Abellán and Moral [1] for the additivity requirement for the maximum entropy on credal sets if we consider Proposition 8.4.1.

Therefore, unlike the other total uncertainty measures on belief intervals for singletons proposed so far, $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ verifies all the essential mathematical properties for this type of measure.

8.4.2 Behavioral requirements of our proposed measure

Now, we analyze the fundamental behavioral requirements for uncertainty measures on belief intervals for singletons, described in Section 8.3, for our proposal $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$.

- **Computational complexity**: The procedure proposed to compute $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ (Algorithm 12) is not as direct as the computation of the other total uncertainty measures on belief intervals for singletons proposed so far. Nevertheless, the computation of $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ is considerably faster than the maximum entropy on the credal set consistent with the BPA m (the well-established total uncertainty measure in ET) because the algorithms proposed so far in the literature for the latter measure work with the whole power set of X, while Algorithm 12 only takes into account the belief and plausibility values for singletons.

- **Coherent disaggregation**: As shown in Equation (8.7), $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ can be decomposed into two measures that quantify conflict and non-specificity.

  The non-specificity part, $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ - $S_*\left(\mathcal{P}(\mathcal{I}_m)\right)$, is equal to 0 if and only if, $\mathcal{P}(\mathcal{I}_m)$ contains a single probability distribution. When all probability distributions on X belong to $\mathcal{P}(\mathcal{I}_m)$, $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$ - $S_*\left(\mathcal{P}(\mathcal{I}_m)\right)$ reaches its maximum value.

  Regarding the conflict part of $S^*\left(\mathcal{P}(\mathcal{I}_m)\right)$, $S_*\left(\mathcal{P}(\mathcal{I}_m)\right)$, it attains its minimum value, 0, when a degenerate probability distribution belongs to $\mathcal{P}(\mathcal{I}_m)$. The maximum value of $S_*\left(\mathcal{P}(\mathcal{I}_m)\right)$ is obtained when $\mathcal{P}(\mathcal{I}_m)$ only contains the uniform probability distribution.

Consequently, we can state that the decomposition of $S^*(\mathcal{P}(\mathcal{I}_m))$ into conflict and non-specificity measures is pretty logical.

- **Sensitivity to changes**: For analyzing the sensitivity of $S^*(\mathcal{P}(\mathcal{I}_m))$ to changes in the belief intervals for singletons, we use the following example, based on the one employed in [21] to analyze the sensitive to changes in the evidence for uncertainty measures on BPAs.

  Example 8.4.2 *Let $X = \{x_1, x_2\}$ be a finite set and m the following BPA on X:*

  $$m(\{x_1\}) = m_1, \quad m(\{x_2\}) = m_2, \quad m(\{x_1, x_2\}) = m_{12} = 1 - m_1 - m_2,$$

  where $0 \leqslant m_i \leqslant 1$, for $i = 1, 2$, and $m_1 + m_2 \leqslant 1$. We have the following set of belief intervals for singletons, $\mathcal{I}_m$:

  $$x_1 \to [m_1, 1 - m_2], \quad x_2 \to [m_2, 1 - m_1].$$

  It should be noted that the width of both intervals is equal to $1 - m_1 - m_2 = m_{12}$. Thus, the non-specificity value is determined by means of m_{12}. The conflict value depends on the interaction of m_1 and m_2 (recall that the conflict value of a set of belief intervals for singletons is determined via the interaction between the belief and plausibility values). Without loss of generality, we assume that the value of m_1 is known. We distinguish two cases:

  - ***Case 1:*** $m_1 \geqslant 0.5$*: It holds that* $m_2 \leqslant 0.5 \leqslant m_1 \Rightarrow S^*(\mathcal{P}(\mathcal{I}_m)) = S(m_1, 1 - m_1), \quad S_*(\mathcal{P}(\mathcal{I}_m)) = S(m_2, 1 - m_2),$
    $(S^* - S_*)(\mathcal{P}(\mathcal{I}_m)) = S(m_1, 1 - m_1) - S(m_2, 1 - m_2)$[4].

    The amount of total uncertainty keeps constant. The conflict part increases as m_2 is greater, which is logical if we take into account that $m_2 \leqslant 0.5 \leqslant m_1$. The non-specificity value increases when m_2 decreases or, equivalently, when m_{12} increases. Remark that the non-specificity value of $\mathcal{I}_m$ depends on m_{12}. Hence, we can state that the variations of the conflict and non-specificity values of $S^*(\mathcal{P}(\mathcal{I}_m))$ as m_2 changes are pretty coherent.

  - ***Case 2:*** $m_1 < 0.5$*. In this case,*

    $$S^*(\mathcal{P}(\mathcal{I}_m)) = S(\alpha_2, 1 - \alpha_2), \quad S_*(\mathcal{P}(\mathcal{I}_m)) = S(\alpha, 1 - \alpha),$$

    where $\alpha_2 = \max(m_2, 0.5), \quad \alpha = \min(m_1, m_2)$*. Consequently, the conflict value depends on the minimum value between m_1 and m_2, which is very logical.*

    For the non-specificity part, three cases are distinguished:

[4] Within this example, $S(a, 1 - a)$ with $a \in [0, 1]$ denotes the Shannon entropy of the probability distribution p_a defined as $p(x_1) = a, \quad p(x_2) = 1 - a$.

1. $m_2 \leqslant m_1 \leqslant 0.5$. *In such a case:*

   $$S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(0.5, 0.5), \quad S_*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(m_2, 1 - m_2),$$

   $$\left(S^* - S_*\right)\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(0.5, 0.5) - S(m_2, 1 - m_2).$$

   We may observe that the total uncertainty value keeps constant and the non-specificity value decreases as m_2 *increases (*m_{12} *decreases), which makes a lot of sense.*

2. $m_1 \leqslant m_2 \leqslant 0.5$. *Then,* $S_*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(m_1, 1 - m_1)$, *which implies that the conflict part does not vary. In addition,* $S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(0.5, 0.5)$. *Therefore, the total uncertainty, conflict, and non-specificity values keep constant. It could be considered an undesirable behavior. Nonetheless, in this situation, since* $1 - m_i > 0.5$, *for* $i = 1, 2$, *it might make sense to considered a total uncertainty value as the plausibility of each singleton is greater than* 0.5.

3. $m_1 < 0.5 \leqslant m_2$. *In this case,*

   $$S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(m_2, 1 - m_2), \quad S_*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(m_1, 1 - m_1),$$

   $$\left(S^* - S_*\right)\left(\mathcal{P}\left(\mathcal{I}_m\right)\right) = S(m_2, 1 - m_2) - S(m_1, 1 - m_1).$$

   The conflict part does not vary and the non-specificity value decreases as m_2 *is higher, i.e,* m_{12} *is lower. This is quite coherent.*

From the previous example, we can conclude that $S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$ is sensitive to changes in the belief intervals for singletons, directly or through its parts of conflict and non-specificity.

In this way, it could be stated that, unlike the total uncertainty measures on belief intervals for singletons proposed so far, $S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$ satisfies all the crucial behavioral requirements for this kind of measure, although its computation is more complex.

Moreover, it should be noted that the maximum entropy on the credal set consistent with the set of belief intervals for singletons is always greater or equal than the maximum entropy on the credal set compatible with the associated belief function, as the following proposition shows:

Proposition 8.4.4 *Let* m *be a BPA on* X *and* Bel_m *its associated belief function. Let* $\mathcal{I}_m$ *be the set of belief intervals for singletons corresponding to* m. *Let* $\mathcal{P}\left(\mathrm{Bel}_m\right)$ *and* $\mathcal{P}\left(\mathcal{I}_m\right)$ *denote the credal sets compatible with* Bel_m *and* $\mathcal{I}_m$, *respectively. It always holds that:*

$$S^*\left(\mathcal{P}\left(\mathrm{Bel}_m\right)\right) \leqslant S^*\left(\mathcal{P}\left(\mathcal{I}_m\right)\right)$$

The proof of this result is trivial taking into account that it is always satisfied that $\mathcal{P}(Bel_m) \subseteq \mathcal{P}(\mathcal{I}_m)$. Hence, our proposed measure provides an upper bound of the maximum of entropy on the credal set associated with a belief function, the well-established total uncertainty measure in ET. In addition, $S_*(\mathcal{P}(\mathcal{I}_m)) \leqslant S_*(\mathcal{P}(Bel_m))$. In consequence, the conflict value provided by our uncertainty measure is always lower or equal than the conflict value captured by $S_*(\mathcal{P}(Bel_m))$. In contrast, the non-specificity value of $S^*(\mathcal{P}(\mathcal{I}_m))$ is always greater or equal than the non-specificity value of $S_*(\mathcal{P}(Bel_m))$. It makes sense because the main difference between uncertainty in ET and probability theory resides in the non-specificity part, and our proposed measure enhances this idea.

8.5 Computation of uncertainty measures on the A-NPI-M

Let X be a discrete variable whose set of possible values is $\{x_1, x_2, \dots, x_t\}$. Suppose that there is a sample of N independent and identically distributed observations about X. For each i = $1, 2, \dots, t$, let $n(x_i)$ denote the number of observations of x_i in the sample. Let t_{obs} (t_{unobs}) be the number of observed (unobserved) values of X in the sample.

$$t_{obs} = |\{x_i \mid n(x_i) > 0, \quad i = 1, 2, \dots, t\}|,$$
$$t_{unobs} = |\{x_i \mid n(x_i) = 0, \quad i = 1, 2, \dots, t\}|.$$

In this section, we show how to compute the main uncertainty measures on the A-NPI-M credal set on X, $\mathcal{P}(\mathcal{I}_{ANPI})$, determined via Equation (2.69).

8.5.1 Maximum entropy

Algorithm 4 shows the procedure proposed in [5] to compute the maximum entropy with the A-NPI-M. In this subsection, we express the algorithm a little bit more simple than in that work.

Let $T(i)$ denote the number of values of X observed i times:

$$T(i) = \left|\left\{x_j \mid n(x_j) = i, \quad 1 \leqslant j \leqslant t\right\}\right|. \tag{8.9}$$

The idea is the same as in the algorithm of maximum entropy for reachable probability intervals: the resulting probability distribution has to be as close to the uniform distribution as possible.

We start by assigning the A-NPI-M lower probabilities to each $x_j, \forall j = 1, 2, \dots, t$. Then, the values for which there are no observations or only one have assigned the lowest probability (in both cases, the lower probability is equal to 0). The probability mass to distribute between all values is equal to $\frac{t_{obs}}{N}$. If the number of values for which there are 0 or 1 observations is lower than t_{obs}, then the probability mass is equally distributed between these values. Otherwise, we sum $\frac{1}{N}$ to the probability of each value x_j for which $n(x_j) \in \{0, 1\}$. Then, the non-observed values have assigned the A-NPI-M upper probability and, between the rest of the values, the ones observed 1 or 2 times have assigned the lowest probability. The resting probability mass to distribute is equal to $\frac{t_{obs} - T(0) - T(1)}{N}$. We iteratively repeat the process until the probability mass is completely distributed among the values of X.

In this way, our proposed procedure to obtain the probability distribution of maximum entropy on an A-NPI-M credal set is given in Algorithm 14.

8.5.2 Minimum entropy

A-NPI-M probability intervals are reachable probability intervals and, thus, Choquet capacities of order 2. In consequence, the algorithm proposed by Abellán and Moral [15] to compute the minimum entropy for Choquet capacities of order 2 could be employed. However, due to the special structure of A-NPI-M probability intervals, the probability distribution of minimum entropy with the A-NPI-M can be obtained in a very quick way via Lemma 8.4.1.

The following theorem shows how to obtain the probability distribution of minimum entropy on an A-NPI-M credal set:

Theorem 8.5.1 *Let $(n_1^*, n_2^*, \dots, n_t^*)$ be the array of observed frequencies ordered in a decreasing way. The probability distribution of minimum entropy on $\mathcal{P}(\mathcal{I}_{ANPI})$ is the one $\underline{p}_{ANPI}$ that satisfies*

$$\underline{p}^*_{ANPI} = \left(\frac{n_1^*+1}{N}, \dots, \frac{n^*_{\frac{t_{obs}-1}{2}}+1}{N}, \frac{n^*_{\frac{t_{obs}+1}{2}}}{N}, \frac{n^*_{\frac{t_{obs}+1}{2}+1}-1}{N}, \dots, \frac{n^*_{t_{obs}}-1}{N}, 0, \dots, 0 \right)$$

if t_{obs} is odd,

$$\underline{p}^*_{ANPI} = \left(\frac{n_1^*+1}{N}, \dots, \frac{n^*_{\frac{t_{obs}}{2}}+1}{N}, \frac{n^*_{\frac{t_{obs}}{2}+1}-1}{N}, \dots, \frac{n^*_{t_{obs}}-1}{N}, 0, \dots, 0 \right) \text{ if } t_{obs} \text{ is even,}$$

where $\underline{p}^*_{ANPI}$ denotes the array of $\underline{p}_{ANPI}$ decreasingly ordered.

Algorithm 14: Proposed procedure to compute the probability distribution of maximum entropy with the A-NPI-M.

```
Procedure Determine probability distribution of maximum entropy with the A-NPI-M(Observed frequencies in the sample $n(x_1), n(x_2), \dots, n(x_t)$)
for $j = 1$ to $t$ do
    if $n(x_j) \leqslant 1$ then
        $\hat{p}^{ANPI}(x_j) \leftarrow 0$
    else
        $\hat{p}^{ANPI}(x_j) \leftarrow \frac{n(x_j)-1}{N}$
$mass \leftarrow t_{obs}$
$i \leftarrow 0$
while $mass > 0$ do
    if $T(i) + T(i+1) < mass$ then
        for $j = 1$ to $t$ do
            if $n(x_j) \in \{i, i+1\}$ then
                $\hat{p}^{ANPI}(x_j) \leftarrow \hat{p}^{ANPI}(x_j) + \frac{1}{N}$
                $mass \leftarrow mass - 1$
    else
        for $j = 1$ to $t$ do
            if $n(x_j) \in \{i, i+1\}$ then
                $\hat{p}^{ANPI}(x_j) \leftarrow \hat{p}^{ANPI}(x_j) + \frac{mass}{N(T(i)+T(i+1))}$
        $mass \leftarrow 0$
    $i \leftarrow i + 1$
return $\hat{p}^{ANPI}$
```

Proof: Suppose that t_{obs} is odd. Let q be a probability distribution belonging to $\mathcal{P}(\mathcal{I}_{ANPI})$ and q^* its corresponding array ordered decreasingly. Then:

$$\sum_{i=1}^{j} q_i^* \leqslant \sum_{i=1}^{j} \frac{n_i^*+1}{N} = \sum_{i=1}^{j} \left(\underline{p}^*_{ANPI}\right)_i, \quad \forall j = 1, \dots, \frac{t_{obs}-1}{2}.$$

For $j = \frac{t_{obs}+1}{2}, \dots, t_{obs}-1$, we have that:

$$\sum_{i=1}^{j} q_i^* = 1 - \sum_{i=j+1}^{t} q_i^* \leqslant 1 - \sum_{i=j+1}^{t} max\left(0, \frac{n_i^*-1}{N}\right) = 1 - \sum_{i=j+1}^{t_{obs}} \frac{n_i^*-1}{N}$$
$$= 1 - \sum_{i=j+1}^{t_{obs}} \left(\underline{p}^*_{ANPI}\right)_i = 1 - \sum_{i=j+1}^{t} \left(\underline{p}^*_{ANPI}\right)_i = \sum_{i=1}^{j} \left(\underline{p}^*_{ANPI}\right)_i.$$

Obviously:

$$\sum_{i=1}^{j} \left(\underline{p}^*_{ANPI}\right)_i = 1 \geqslant \sum_{i=1}^{j} q_j^*, \quad \forall j = t_{obs}, \dots, t.$$

To sum up,

$$\sum_{i=1}^{j} q_i^* \leqslant \sum_{i=1}^{j} \left(\underline{p}^*_{ANPI}\right)_i, \quad \forall j = 1, 2, \dots, t,$$

and Lemma 8.4.1 allows us to conclude that $S(q) \geqslant S\left(\underline{p}_{ANPI}\right)$. Therefore, $\underline{p}_{ANPI}$ is the probability distribution of minimum entropy on $\mathcal{P}(\mathcal{I}_{ANPI})$.

The proof in the case that t_{obs} is even is identical.

□

8.5.3 Generalized Hartley measure

Let m^{ANPI} denote the Möbius inverse associated with the A-NPI-M coherent lower probability function. The Generalized Hartley measure (GH) is calculated by means of the following formula:

$$GH^{ANPI} = \sum_{A \subseteq \{x_1, x_2, \dots, x_t\}} m^{ANPI}(A) \log_2(|A|). \tag{8.10}$$

As shown in Proposition 7.3.4, the Möbius inverse for a set whose cardinality is greater or equal than 2 depends on the cardinality of the set, the number of observed values in the set, and the sample size (N). Moreover, the number of sets with a certain cardinality and a determinate number of observed values depends on the total number of values (t) and how many of them have been observed (t_{obs}). Thereby, the GH value for A-NPI-M credal sets depends on t, t_{obs}, and N.

Firstly, we may observe that, according to Corollary 7.3.2, for a set of cardinality greater than 1 but lower or equal than $\frac{t}{2}$, the Möbius inverse is equal to 0. The Möbius inverse for singletons is not necessarily equal to 0. Nonetheless, as it is well known, $\log_2 1 = 0$. So, the singletons do not influence the calculation of GH. Thus, only the sets whose cardinality is greater than $\frac{t}{2}$ influence the computation of GH^{ANPI}.

Secondly, for a set A whose cardinality is c_A, the number of observed values in the set, t^A_{obs}, clearly verifies that $t^A_{obs} \leqslant t_{obs}$ and $t^A_{obs} \leqslant c_A$. Furthermore, the number of observed values in the set is greater or equal than $t_{obs} - |\overline{A}| = t_{obs} - t + c_A$ (this last number can be negative). Consequently, $\min(t_{obs}, c_A) \geqslant t^A_{obs} \geqslant \max(0, t_{obs} - t + c_A)$.

For determining the number of sets whose cardinality is c_A and have t^A_{obs} observed values, we think as follows: we can choose the t^A_{obs} observed values in $\binom{t_{obs}}{t^A_{obs}}$ possible ways. Likewise, the number of ways in which the $c_A - t^A_{obs}$ non-observed values can be chosen is equal to $\binom{t_{unobs}}{c_A - t^A_{obs}}$. Therefore, the number of sets with cardinality c_A and t^A_{obs} observed values is equal to $\binom{t_{obs}}{t^A_{obs}} \times \binom{t_{unobs}}{c_A - t^A_{obs}}$.

Considering the previous points and the expression given in Proposition 7.3.4 to calculate the Mobius inverse for a set whose cardinality is greater than 1, the procedure for the computation of GH^{ANPI} is given in Algorithm 15.

8.5.4 Examples

In this subsection, we show two examples about the computation of the uncertainty measures considered in this section with the A-NPI-M.

Example 8.5.1 *Let* X *be a variable that takes values in* $\{x_1, x_2, x_3, x_4\}$. *Let* $(n_1, n_2, n_3, n_4) = (7, 2, 1, 8)$ *be the array of observed frequencies. In this case,* $N = n_1 + n_2 + n_3 + n_4 = 18$, $t = t_{obs} = 4$. *The set of A-NPI-M probability intervals is:*

$$\left\{ \left[\frac{6}{18}, \frac{8}{18}\right]; \left[\frac{1}{18}, \frac{3}{18}\right]; \left[0, \frac{2}{18}\right]; \left[\frac{7}{18}, \frac{9}{18}\right] \right\}.$$

Let $\hat{p}^{ANPI} = \left(\hat{p}^{ANPI}(x_1), \hat{p}^{ANPI}(x_2), \hat{p}^{ANPI}(x_3), \hat{p}^{ANPI}(x_4)\right)$ *denote the array of the probability distribution of maximum entropy on the corresponding credal set.*

Algorithm 15: Procedure to compute the Möbius inverse with the A-NPI-M.

```
Procedure Determine Möbius inverse associated with the
A-NPI-M(Observed frequencies in the sample $(n(x_1), n(x_2), \dots, n(x_t))$)
$GH^{ANPI} \leftarrow 0$
if $t$ is odd then
    $min\_c_A \leftarrow \frac{t+1}{2}$
else
    $min\_c_A \leftarrow \frac{t}{2} + 1$
for $c_A = min\_c_A$ to $t$ do
    if $t_{obs} \leqslant c_A$ then
        $max\_t^A_{obs} \leftarrow t_{obs}$
    else
        $max\_t^A_{obs} \leftarrow c_A$
    if $t_{obs} - t + c_A > 0$ then
        $min\_t^A_{obs} \leftarrow t_{obs} - t + c_A$
    else
        $min\_t^A_{obs} \leftarrow 0$
    $sum\_inverses\_c_A \leftarrow 0$
    for $t^A_{obs} = min\_t^A_{obs}$ to $max\_t^A_{obs}$ do
        $sum\_inverses\_t^A_{obs} \leftarrow 0$
        for $i = 1$ to $t^A_{obs}$ do
            $sum\_inverses \leftarrow 0$
            for $j = 0$ to $c_A - t^A_{obs}$ do
                if $2i - t + j > 0$ then
                    $sum\_inverses \leftarrow$
                     $sum\_inverses + \binom{c_a - t^A_{obs}}{j} \times (-1)^{c_a - i - j} \times (2i - t + j)$
            $sum\_inverses\_t^A_{obs} \leftarrow$
             $sum\_inverses\_t^A_{obs} + sum\_inverses \times \binom{t^A_{obs}}{i}$
        $sum\_inverses\_c_A \leftarrow$
         $sum\_inverses\_c_A + sum\_inverses\_t^A_{obs} \times \binom{t\_obs}{t^A_{obs}} \times \binom{t_{unobs}}{c_A - t^A_{obs}}$
    $GH \leftarrow GH + \log_2 c_A \times sum\_inverses\_c_A$
$GH^{ANPI} \leftarrow \frac{GH^{ANPI}}{N}$
return $GH^{ANPI}$
```

If we utilize Algorithm 14, the array $\hat{p}^{ANPI}$ *initially has the values* $\hat{p}^{ANPI} = \left(\frac{6}{18}, \frac{1}{18}, 0, \frac{7}{18}\right)$. *This array takes the following values after each one of the corresponding iterations of the loop:*

$$i = 0 \to \hat{p}^{ANPI} = \left(\frac{6}{18}, \frac{1}{18}, \frac{1}{18}, \frac{7}{18}\right),$$

$$i = 1 \to \hat{p}^{ANPI} = \left(\frac{6}{18}, \frac{2}{18}, \frac{2}{18}, \frac{7}{18}\right),$$

$$i = 2 \to \hat{p}^{ANPI} = \left(\frac{6}{18}, \frac{3}{18}, \frac{2}{18}, \frac{7}{18}\right).$$

The array obtained when $i = 2$, $\hat{p}^{ANPI} = \left(\frac{6}{18}, \frac{3}{18}, \frac{2}{18}, \frac{7}{18}\right)$, *is the probability distribution that reaches the maximum entropy on the A-NPI-M credal set.*

According to Theorem 8.5.1, the probability distribution that attains the minimum entropy on the A-NPI-M credal set, $\underline{p}_{ANPI}$, *is given by:*

$$\underline{p}_{ANPI} = \left(\frac{n_2^* + 1}{N}, \frac{n_3^* - 1}{N}, \frac{n_4^* - 1}{N}, \frac{n_1^* + 1}{N}\right) = \left(\frac{8}{18}, \frac{1}{18}, 0, \frac{9}{18}\right).$$

The value of the generalized Hartley measure, obtained through Algorithm 15, is equal to:

$$GH^{ANPI} = \frac{1}{18} \times (8 \log_2 3 - 4 \log_2 4).$$

Example 8.5.2 *Suppose again that X is a variable whose set of possible values is* $\{x_1, x_2, x_3, x_4\}$. *Let* $(n_1, n_2, n_3, n_4) = (0, 3, 9, 1)$ *denote the array of observed frequencies. In this case,* $N = 13$, $t = 4$, $t_{obs} = 3$. *We have the following set of A-NPI-M probability intervals:*

$$\left\{\left[0, \frac{1}{13}\right]; \left[\frac{2}{13}, \frac{4}{13}\right]; \left[\frac{8}{13}, \frac{10}{13}\right]; \left[0, \frac{2}{13}\right]\right\}.$$

Let $\hat{p}^{ANPI}$ *denote the array of the probability distribution of maximum entropy on the associated credal set. If we apply Algorithm 14, it initially has the values* $\hat{p}^{ANPI} = \left(\hat{p}^{ANPI}(x_1), \hat{p}^{ANPI}(x_2), \hat{p}^{ANPI}(x_3), \hat{p}^{ANPI}(x_4)\right) = \left(0, \frac{2}{13}, \frac{8}{13}, 0\right)$. *It takes the following values after each one of the corresponding iterations of the loop.*

$$i = 0 \to \hat{p}^{ANPI} = \left(\frac{1}{13}, \frac{2}{13}, \frac{8}{13}, \frac{1}{13}\right),$$

$$i = 1 \to \hat{p}^{ANPI} = \left(\frac{1}{13}, \frac{2}{13}, \frac{8}{13}, \frac{2}{13}\right).$$

The probability distribution obtained when $i = 1$, $\hat{p}^{ANPI} = \left(\frac{1}{13}, \frac{2}{13}, \frac{8}{13}, \frac{2}{13}\right)$, *is the one that attains the maximum entropy on the A-NPI-M credal set.*

In order to obtain the probability distribution of minimum entropy on the A-NPI-M credal set, $\underline{p}_{ANPI}$, *Theorem 8.5.1 is applied:*

$$\underline{p}_{ANPI} = \left(0, \frac{n_2^*}{N}, \frac{n_1^*+1}{N}, \frac{n_3^*-1}{N}\right) = \left(0, \frac{3}{13}, \frac{10}{13}, 0\right).$$

The value of the generalized Hartley measure, computed via Algorithm 15, is equal to:

$$GH^{ANPI} = \frac{1}{13} \times \left[5 \log_2 3 - 2 \log_2 4\right].$$

8.6 Concluding remarks

The study of uncertainty measures in imprecise probabilities has its origin in the study of uncertainty measures in Evidence theory (ET). Within this theory, the maximum entropy is the well-established uncertainty measure as it is the only one that satisfies all essential mathematical properties and behavioral requirements. Nevertheless, this measure has an important drawback: its computation is notably complex.

For this reason, many alternatives to the maximum entropy have been developed during the last years. Among such alternatives, one of the most known is Deng entropy. In previous works, it was proved that this measure violates most of the crucial mathematical properties for uncertainty measures in ET, and its behavior in some scenarios is questionable. In order to solve some shortcomings of Deng entropy, two modifications of this measure were proposed a few years ago. In this chapter, we have demonstrated that these modifications also violate most of the fundamental mathematical properties for uncertainty measures in ET. Moreover, we have also shown that, in some scenarios, the behavior of the modifications of the Deng entropy is questionable. For example, in both modifications, the conflict part may be positive when all focal elements are not disjunct. Furthermore, the extension of these modified Deng entropies to more general theories than ET is not trivial.

As another alternative to the maximum entropy, belief intervals for singletons have been commonly employed during the last years for quantifying uncertainty-based information in ET. Indeed, they are easier to manage than BPAs for representing uncertainty-based information. In this chapter, we have carried out a study about the fundamental mathematical properties and behavioral requirements for total uncertainty measures on belief intervals for singletons. Such a study has been based on the one previously carried out for total uncertainty measures on BPAs. It has been highlighted that, when the belief intervals for singletons are reduced to a single probability distribution, a total uncertainty measure on such intervals must coincide with the one well-established in classical probability theory, i.e the Shannon entropy; if it is only known that the uncertainty-based information expressed via the belief intervals for singletons is focused on a subset of alternatives, then a total uncertainty measure must take the form of an increasing function with respect to the cardinality of that subset; the range of a total uncertainty measure on belief intervals for singletons must be coherent: the minimum value, which has to be equal to 0, must be attained if, and only if, the information is focused on a singleton, and its maximum value when all probability distributions are consistent with the belief intervals fro singletons; a total uncertainty measure on belief intervals for singletons has to be consistent with an increase or decrease of information expressed by such intervals; the values of a total uncertainty measure on the set of belief intervals for singletons corresponding to a BPA defined over a joint space that can be decomposed on more simple sets must be coherent. Our proposed set of behavioral requirements for total uncertainty measures on belief intervals for singletons reveals that the computation of a measure of this type must not be too complex; it has to be possible to separate a total uncertainty measure on belief intervals for singletons into two ones that coherently indicate conflict and non-specificity, respectively; a total uncertainty measure on belief intervals for singletons must be sensitive to changes in these intervals, directly or via conflict and non-specificity. We have shown that none of the uncertainty measures on belief intervals for singletons proposed so far satisfies all the crucial mathematical properties and behavioral requirements for this kind of measure.

Furthermore, we have proposed a total uncertainty measure on belief intervals for singletons that consists of the maximum entropy on the credal set compatible with such intervals. We have demonstrated that, even though our proposed measure requires a more complex computation than the other uncertainty measures on belief intervals for singletons proposed so far, it is the only one that satisfies all the essential mathematical properties and behavioral requirements for uncertainty measures on belief intervals for singletons. We have also highlighted that the maximum entropy on the belief intervals for singletons is always greater or equal than the maximum entropy on the credal set associated with a belief function. Thereby, our proposal gives an upper bound of the maximum entropy, the well-established uncertainty measure in ET, the computation of the former measure being considerably simpler than the latter.

Concerning imprecise probability models, in this chapter, we have shown how to compute the most important uncertainty measures for credal sets associated with the Approximate Non-Parametric Predictive Inference Model (A-NPI-M). Remark that this model can be expressed by means of a reachable set of probability intervals and, unlike the IDM, the A-NPI-M does not assume previous knowledge about the data through a parameter. Specifically, we have presented a little bit more simple algorithm than the one proposed so far to calculate the maximum entropy (the well-established total uncertainty measure on credal sets) with the A-NPI-M; we have proposed and proved a result that allows us to quickly obtain the minimum of entropy on A-NPI-M credal sets; in addition, we have shown how to calculate the generalized Hartley measure with the A-NPI-M. Hence, with this model, it is possible to know how both types of uncertainty: conflict (the difference between the maximum entropy and the generalized Hartey measure, or the minimum of entropy), and non-specificity (the generalized Hartley measure or the difference between the maximum and minimum entropy), coexist. Those procedures represent useful tools to make the A-NPI-M very suitable to be employed in practical applications.

9 PRECISE CLASSIFICATION WITH NOISE

9.1 Introduction

Nowadays, the classification task is widely used in many domains. It consists of learning a model that predicts, for a given instance described via a set of attributes or features, the value of a class variable. Many algorithms for classification have been developed so far. Clearly, the performance of classification methods is worsened when there is noise in the data, i.e. when data contain errors.

Naïve Bayes (NB) [85] is one of the most simple approaches to traditional classification. It assumes that all attributes are independent given the class variable, which is not often realistic. Despite this unrealistic assumption, NB has obtained good results in practice, comparable with more sophisticated classification methods, especially when the attributes are not strongly correlated [82, 93, 133]. Moreover, the NB algorithm, as a consequence of its independence assumption, is much faster than other more sophisticated classification models and the required computational cost is significantly lower.

The NB classifier is based on the Bayes formula [31]. In NB, this formula is used naïvely, i.e, assuming the independence condition. As pointed out by Cestnik [47, 48], the evaluation of the naïve Bayesian formula is pretty influenced by the estimation of the conditional probabilities. In Section 4.4, we have shown that the classical probability estimation through relative frequencies has important shortcomings; the Laplace estimation [103], which was proposed to solve such shortcomings, has some drawbacks too; Cestnik [47, 48] introduced a new probability estimation (m-probability estimation) that considers the prior probabilities of the class values when the conditional probability of the class values given the value of an attribute are estimated. Cestnik [47] experimentally showed that m-probability estimation provides better results than the previous approaches of conditional probability estimation, although the experimentation carried out was very scarce, using only four databases without added noise. In addition, Cestnik [48] experimentally showed that m-probability estimation in the tree-pruning process improves the results of standard pruning methods.

In spite of the improvement in the performance of the NB model with the m-probability estimation, the algorithm is still quite sensitive to noise. This happens because the estimation of the prior probabilities is still done by means of relative frequencies with Laplacian correction, which is clearly deteriorated with the presence of noise. For this reason, in this chapter, we use the Imprecise Dirichlet Model (IDM) to estimate the prior probabilities of the class values. As pointed out previously, this imprecise model has been tested to be useful for improving the performance of standard models when there is noise in the data. An example of this issue is the Credal C4.5 algorithm [149].

Specifically, in this chapter, a new Naïve Bayes model, called the Imprecise m-probability-estimation Naïve Bayes (ImNB), is proposed. It combines the m-probability estimation with the IDM to obtain a classifier less sensitive to class noise. An extensive experimental analysis is carried out where our new NB approach is compared with NB using the m-probability, the Laplace, and classical probability estimations. The mentioned algorithms are applied to many datasets without noise and with different levels of added class noise. The experimental analysis shows that the Cestnik proposal obtains much better results than NB with Laplace and classical estimations of probabilities, with much more exhaustive experimentation than in [47] and that the proposed method performs better than the Cestnik model.

This chapter is organised in the following way: Section 9.2 describes our proposed Imprecise m-probability estimation Naive Bayes. The experimental study carried out in this chapter is detailed in Section 9.3. Section 9.4 concludes this chapter.

9.2 Imprecise m-probability-estimation Naïve Bayes

Let C be the class variable and $\Omega_C = \{c_1, c_2, \ldots, c_K\}$ its set of possible values. Let $\{X^1, X^2, \ldots, X^d\}$ be the set of predictive attributes. Within this section, we assume that the domain of each attribute is finite, that is, the possible values of X^i are $\{x^i_1, x^i_2, \ldots, x^i_{t_i}\}, \quad \forall i = 1, 2, \ldots, d$.

Our proposed Imprecise m-probability estimation Naïve Bayes model (ImNB) uses the naïve assumption (see Equation (4.10)). In this way, in order to classify an instance with attribute vector $\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d})$, where $r_i \in \{1, 2, \ldots, t_i\} \quad \forall i = 1, 2, \ldots, d$, ImNB predicts the following class value:

$$h(\mathbf{x})^{NB} = \arg \max_{c_j \in \Omega_C} P(C = c_j) \prod_{i=1}^{d} \frac{P(C = c_j \mid X^i = x^i_{r_i})}{P(C = c_j)}.$$

The difference between ImNB and the other NB algorithms resides in the estimation of the probabilities $P(C = c_j)$ and $P(C = c_j \mid X^i = x^i_{r_i}), \quad \forall j = 1, 2, \ldots, K, \quad r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

Let N_{tr} denote the number of training instances and $n_{tr}(c_j)$ the number of instances in the training set for which $C = c_j, \quad \forall j = 1, 2, \ldots, K$. Let us consider the IDM credal set on C on the training set:

$$\mathcal{P}^{IDM}_{tr}(C) = \left\{ p \in \mathcal{P}(C) \mid \frac{n_{tr}(c_j)}{N_{tr} + s} \leqslant p(c_j) \leqslant \frac{n_{tr}(c_j) + s}{N_{tr} + s}, \quad \forall j = 1, 2, \ldots, K \right\}, \tag{9.1}$$

where s is the IDM parameter and $\mathcal{P}(C)$ the set of all probability distributions on C.

Uncertainty measures can be applied to this credal set. As explained before, the maximum entropy is a well-established uncertainty measure on credal sets as it satisfies the required properties. Hence, for the estimation of the prior probabilities, ImNB uses the probability distribution attains the maximum entropy on $\mathcal{P}^{IDM}_{tr}(C)$, namely $\hat{p}_{ImNB}$. Algorithm 16 shows the procedure to obtain such a probability distribution. It is based on Algorithm 3, the procedure proposed so far for the maximum entropy on reachable probability intervals (IDM probability intervals are always reachable).

For the estimation of the conditional probabilities $P(C = c_j \mid X^i = x^i_{r_i}), \quad \forall j = 1, 2, \ldots, K, \quad r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$, ImNB takes the Cestnik model as a reference in the sense that it considers the prior probabilities. However, unlike the Cestnik model, the prior probabilities are estimated through Algorithm 16. According to Cestnik [47], the value of the m parameter in his model should be greater as there is more noise in the data, as happens with the IDM parameter s. For this reason, the same values for s and m are chosen.

In this way, ImNB estimates the conditional probabilities as follows:

$$\hat{P}_{ImNB}(C = c_j \mid X^i = x^i_{r_i}) = \frac{n_{tr}(x^i_{r_i,j}) + s\hat{p}_{ImNB}(c_j)}{n_{tr}(x^i_{r_i}) + s}, \tag{9.2}$$

$n_{tr}(x^i_{r_i,j})$ being the number of training instances that satisfy $X^i = x^i_{r_i} \wedge C = c_j$ and $n_{tr}(x^i_{r_i})$ the number of training instances for which $X^i = x^i_{r_i}, \quad \forall j = 1, 2, \ldots, K, \quad r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

Therefore, for classifying an instance with attribute vector $\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \ldots, x^d_{r_d})$, where $r_i \in \{1, 2, \ldots, t_i\} \quad \forall i = 1, 2, \ldots, d$, ImNB makes the following prediction:

$$h(\mathbf{x})^{ImNB} = \arg\max_{c_j \in \Omega_C} \hat{p}_{ImNB}(c_j) \prod_{i=1}^{d} \frac{\hat{P}_{ImNB}(C = c_j \mid X^i = x^i_{r_i})}{\hat{p}_{ImNB}(c_j)}, \tag{9.3}$$

Algorithm 16: Procedure to compute the probability distribution of maximum entropy on the IDM credal set on the training set.

```
Procedure Determine probability distribution of maximum entropy with the IDM on the training set (Number of training instances $N_{tr}$, class frequencies in the training set $(n_{tr}(c_1), n_{tr}(c_2), \dots, n_{tr}(c_K))$, IDM parameter s)
$s' \leftarrow s$
for $j = 1$ to $K$ do
    $n'(c_j) \leftarrow n_{tr}(c_j)$
while $s' > 0$ do
    $s'' \leftarrow \min\{s', 1\}$
    $num\_min \leftarrow \left|\{c_j \mid n'(c_j) = \min_{k=1,2,\dots,K} n'(c_k)\}\right|$
    for $j = 1$ to $K$ do
        if $n'(c_j) = \min_{k=1,2,\dots,K}\{n'(c_k)\}$ then
            $n'(c_j) \leftarrow n'(c_j) + \frac{s''}{num\_min}$
    $s' \leftarrow s' - 1$
for $j = 1$ to $K$ do
    $\hat{p}_{ImNB} = \frac{n'(c_j)}{N_{tr}+s}$
return $\hat{p}_{ImNB}$
```

where $\hat{p}_{ImNB}$ is the probability distribution of maximum entropy on the IDM credal set on the training set, obtained through Algorithm 16, and $\hat{P}_{ImNB}(C = c_j \mid X^i = x^i_{r_i})$ is computed via Equation (9.2), $\forall j = 1, 2, \ldots, K, \quad i = 1, 2, \ldots, d.$

We must remark the following issues about ImNB:

- For the estimation of the prior probabilities, ImNB considers the probability distribution that attains the maximum entropy on the IDM credal set, while the Cestnik model uses Laplace's estimation. Thus, the Cestnik model is more sensitive to the presence of class noise in the training data. As pointed out before, imprecise probability models have obtained better results than classical estimators in classification with class noise.

- As the Cestnik model, ImNB takes the prior probabilities of the class values into account for the estimation of the conditional probabilities. In consequence, ImNB also solves the problems that arise with classical and Laplace estimations of such probabilities.

Due to the previous points, it is expected that our proposed ImNB performs better than the other NB methods, explained in Section 4.4.1, the improvement being more notable as there is more class noise in the data. This point is validated in Section 9.3 with exhaustive experimentation.

9.3 Experimentation

9.3.1 Description of the experiments

- **Datasets**: For the experiments, we have selected 75 well-known datasets, obtained from *UCI Machine Learning Repository* [139]. All these datasets have been widely used in the specialized literature for comparing classification algorithms. Tables 9.1 and 9.2 show the most relevant characteristics of each dataset. As can be observed, these datasets are diverse regarding the number of instances, number of continuous and discrete attributes, number of values of the class variable, and number of values of discrete features.

- **Preprocessing**: Since our proposed algorithm only works with discrete attributes, the datasets have been previously discretized. For this purpose, Fayyad and Irani's discretization method [88] has been employed.

- **Algorithms**: In this experimental study, four algorithms have been compared: NB with the classical estimation of probabilities (classical NB), NB using the Laplace smoothing (Laplace NB), NB with the Cestnik model (mNB), and our proposal (ImNB). Classical NB, Laplace NB, and mNB were described in Section 4.4.1, and our proposed ImNB have been presented in Section 9.2.

- **Evaluation**: In order to evaluate the performance of the algorithms considered here, we principally consider the Accuracy metric. In addition, since we are using datasets with class noise, we use the average *Equalized Loss Accuracy* metric (ELA) [181], detailed in Section 4.2.1.2, to evaluate the robustness to class noise of each algorithm considered here.

- **Procedure**: Four noise levels have been considered in our experiments: 0%, 10%, 20%, and 30%. Here, only class noise has been considered. The noise has been added only to training sets. For each dataset, the noise has been added by using a random procedure: given a noise level x, the x% of the instances are randomly selected from the training dataset and their class value is randomly changed to another class value. The instances belonging to the test dataset are left unmodified.

  To compare the results of the classifiers, a 10-fold cross validation procedure has been repeated 10 times for each dataset, level of noise, and algorithm. The same partitions have been used for all methods.

- **Software** and **Parameters**: The Weka software [218] has been employed for our experiments. The implementations available in this software for Classical NB and Laplace NB have been used, and the necessary structures and methods for employing mNB and ImNB have been added. The Weka filters have been utilized for adding noise. Also, for cross-validation, the functionality available in Weka has been used.

  For the ImNB model, in preliminary experiments, it has been noted that the method has a good performance with $m = 4$ as the default value. Obviously, the results would improve if we tuned the value of the parameter m to the level of noise in the data. However, its default value has been used for the experiments. With regard to mNB, Cestnik did not make any recommendation for the m value; we only know that it is related to the level of noise in the data. Hence, a wide range of values for the parameter m has been considered. In concrete, twenty values have been tested for mNB ($m=1, \dots, m=20$) in order to obtain the most appropriate value for each noise level.

- **Statistical evaluation**: Following the recommendations of Demšar [75] for statistical comparisons between the results obtained by three or more methods on many datasets, we have used the Friedman test to compare the performance of the algorithms via the Accuracy metric. If the null hypothesis of this test is rejected, then we compare the performance of the algorithms by using the Holm test. The level of significance used is $\alpha = 0.05$. We present the results of these tests by means of critical diagrams.

For the selection of the m parameter in mNB, the Friedman ranking has been employed. Thereby, the chosen m value is the one that leads to the best rank.

9.3.2 Results and discussion

Table 9.3 shows the Friedman ranks for mNB for each value of the m parameter and each noise level. The best value of the parameter m for each noise level is emphasized using bold font and is the one used for the comparisons of mNB with the other algorithms.

Tables 9.4 and 9.5 present, respectively, the average Accuracy and Friedman ranks of the NB algorithms considered in this experimental analysis with different levels of added noise. In both tables, for each noise level, the best result is marked with bold font and the second-best result with italic font.

Figures 9.1, 9.2 9.3, and 9.4 show the critical diagrams corresponding to the Holm test with 0%, 10%, 20%, and 10% of added noise, respectively. Remark that, in such diagrams, segments are used to connect the algorithms for which there are no statistically significant differences according to the Holm test.

In Table 9.6, the average result obtained by each algorithm in the ELA metric for each noise level can be seen.

From a general point of view, the following points should be noted:

- Both mNB (tuned) and ImNB have better performance than the NB models used as reference (NB with and without Laplace smoothing) on datasets with and without class noise. The improvement is not only with regard to the classifier Accuracy, via the Friedman and Holm tests, but also in terms of robustness to noise (ELA measure).
- We can note that our proposed ImNB obtains better results than Cestnik's approach, this improvement being statistically significant in some cases.

Now, we analyze in detail the experimental results taking into account the following aspects: Average accuracy, Friedman ranks, Holm test, and robustness to noise:

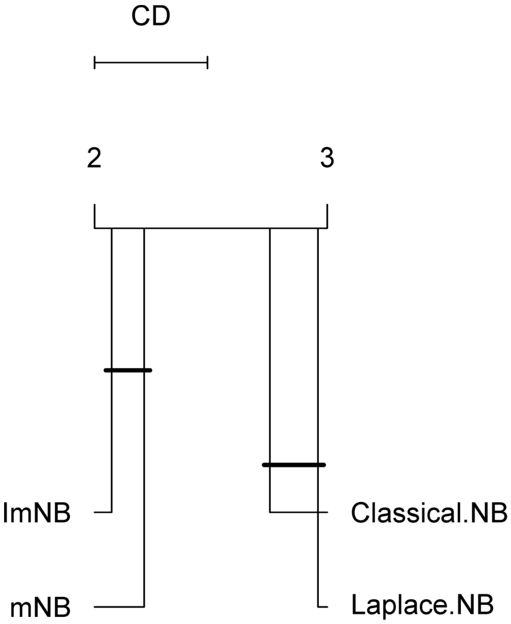

Figure 9.1: Critical diagram about the Accuracy of the NB algorithms on datasets without added noise. CD = critical distance

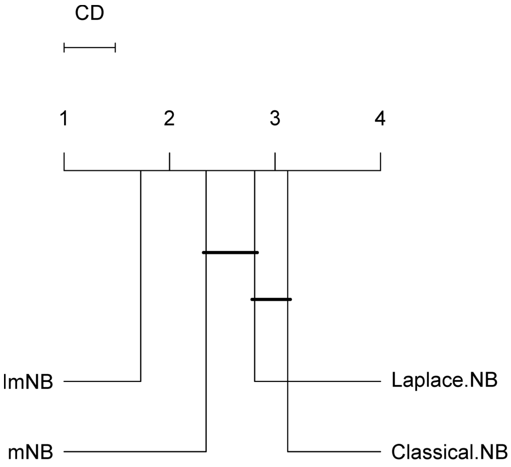

Figure 9.2: Critical diagram about the Accuracy of the NB algorithms on datasets with a 10% of added noise. CD = critical distance

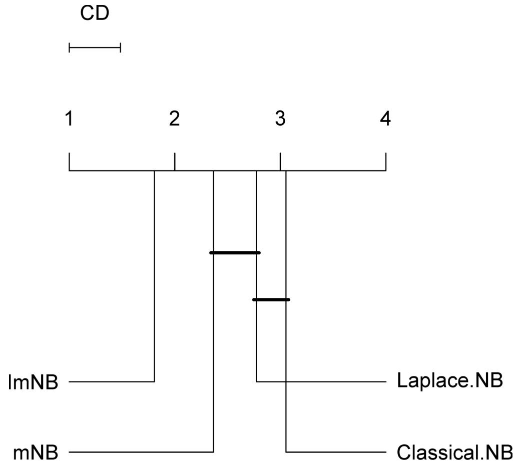

Figure 9.3: Critical diagram about the Accuracy of the NB algorithms on datasets with a 20% of added noise. CD = critical distance

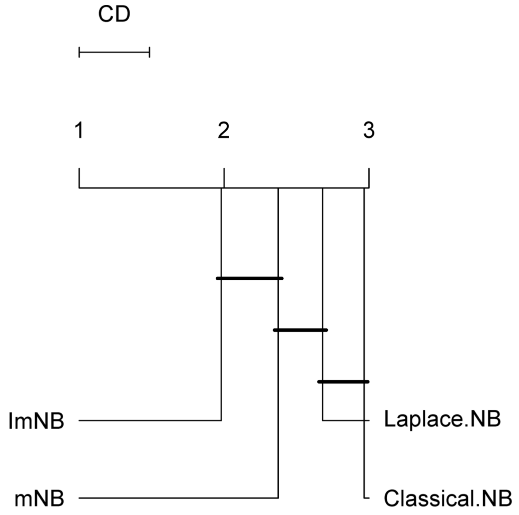

Figure 9.4: Critical diagram about the Accuracy of the NB algorithms on datasets with a 30% of added noise. CD = critical distance

- **Average accuracy**: The methods based on m-probability-estimation, i.e, mNB and ImNB, always attain the highest average Accuracy, regardless of the level of added noise. The best result is always achieved by our proposed method (ImNB), and the second-best result is invariably obtained by Cestnik's approach (mNB) with the tuned parameter. Regarding average Accuracy, the worst result is invariably obtained by the NB without Laplace smoothing (Classical NB). We want to emphasize that this ordering from the best NB model (ImNB) to the worst (Classical NB) occurs independently of the percentage of added noise.

- **Friedman ranks and Holm test**: The outcomes of the statistical tests reinforce the comments made about average Accuracy. The tuned mNB and the ImNB methods obtain the best Friedman ranks for all levels of added noise, and the best classifier in the ranking is always our proposal (ImNB). According to the Friedman ranks, the worst NB model for datasets without added noise is, interestingly, the NB with Laplace smoothing (Laplace NB). Nevertheless, the NB without the Laplacian correction (Classical NB) obtains the worst rank for any level of added noise (10%, 20% or 30%).

  Concerning the Holm test, our approach is the only one that always significantly outperforms the classical NB classifiers with and without Laplace smoothing, regardless of the level of added noise. Indeed, in the critical diagrams, unlike mNB, ImNB is never connected via a segment with Classical NB or Laplace NB. The Cestnik proposal shows a less consistent behavior: it significantly outperforms NB without Laplace smoothing for all noise levels; however, it only performs significantly better than NB with Laplace smoothing when the level of noise is 0%. In the critical diagrams of Figures 9.1 and 9.4, mNB and ImNB are connected via a segment. The contrary happens in the critical diagrams of Figures 9.2 and 9.3. Consequently, for 0% and 30% of added noise, these two algorithms perform equivalently according to the Holm test, whereas ImNB significantly outperforms mNB with the tuned value of the parameter via this test for 10% and 20% of added noise.

- **ELA measure**: According to the average results obtained in this metric, there is no doubt about what methods are the most robust to noise. The ordering obtained in average Accuracy and Friedman's ranking coincides with the sequence obtained in the ELA measure. Therefore, our proposal is in the first place, the tuned mNB classifier in the second, NB with Laplace smoothing in the third place, and the worst result is obtained by NB without Laplace smoothing (classical NB). This outcome occurs independently of the percentage of added noise.

9.3.2.1 *Summary of the results*

With the above analysis, we can summarize the results obtained in this experimental study (Tables 9.3-9.6 and Figures 9.1-9.4) as follows:

- The methods based on m-probability-estimation (mNB and ImNB) outperform the conventional approaches of NB, that is, NB with and without Laplace smoothing (Classical NB and Laplace NB). These outcomes are obtained consistently, regardless of whether the datasets suffer or not from class noise. This happens because, unlike Classical NB and Laplace NB, mNB and ImnB take the prior probabilities into account for this estimation of the conditional probabilities. In this way, mNB and ImNB solve some drawbacks of the classical and Laplace estimations that we commented in Section 4.4.1.

- Considering statistically significant differences, we may notice that the best outcomes are achieved by our proposal. This occurs since, as shown in Section 9.2, ImNB uses the probability distribution of maximum entropy on the IDM credal set for the prior probabilities, while mNB uses the Laplace estimation for such probabilities. In consequence, our proposed ImNB is less sensitive to class noise than mNB.

Table 9.1: Description of the datasets used in our experiments with NB models. Column 'N' is the number of instances, column 'Feat' is the number of features, column 'Num' is the number of numerical features, column 'Discr' is the number of discrete attributes, column 'k' is the number of values of the class variable, and column 'Range' is the range of values of the discrete attributes.

Dataset	N	Feat	Num	Discr	k	Range
acute-infl-nephritis	120	6	1	5	2	2
anneal	898	38	6	32	6	2-10
appendicitis	106	7	7	0	2	-
arrhythmia	452	279	206	73	16	2
audiology	226	69	0	69	24	2-6
autos	205	25	15	10	7	2-22
balance-scale	625	4	4	0	3	-
bank-marketing	4521	16	7	9	2	2-12
banknote-auth	1372	4	4	0	2	-
breast-cancer	286	9	0	9	2	2-13
breast-cancer-wisconsin	699	9	9	0	2	-
bridges-version1	107	11	3	8	6	2-54
bridges-version2	107	11	0	11	6	2-54
bupa	345	6	6	9	2	-
car	1728	6	0	6	4	3-4
cmc	1473	9	2	7	3	2-4
horse-colic	368	22	7	15	2	2-6
credit-rating-australian	690	15	6	9	2	2-14
credit-rating-german	1000	20	7	13	2	2-11
dermatology	366	34	1	33	6	2-4
diabetes-pima	768	8	8	0	2	-
dresses-sales	500	12	1	11	2	5-25
ecoli	366	7	7	0	7	-
fertility-diagnosis	100	9	9	0	2	-
flags	194	29	2	27	8	4-194
glass	214	9	9	0	7	-
glioma16	50	16	16	0	2	-
haberman	306	3	2	1	2	12
heart-disease-cleveland	303	13	6	7	5	2-14
heart-disease-hungarian	294	13	6	7	5	2-14
heart-statlog	270	13	13	0	2	-
hepatitis	155	19	4	15	2	2
hypothyroid	3772	30	7	23	4	2-4
ionosphere	351	35	35	0	2	-
iris	150	4	4	0	3	-
japanese-crx	690	15	6	9	2	2-14
kr-vs-kp	3196	36	0	36	2	2-3
letter	20000	16	16	0	26	-

Table 9.2: Description of the datasets used in our experiments with NB models (cont). Column 'N' is the number of instances, column 'Feat' is the number of features, column 'Num' is the number of numerical features, column 'Discr' is the number of discrete attributes, column 'k' is the number of values of the class variable, and column 'Range' is the range of values of the discrete attributes.

Dataset	N	Feat	Num	Discr	k	Range
liver-disorders	345	6	6	0	2	-
lsvt-voice-rehab	126	310	310	0	2	-
lymphography	146	18	3	15	4	2-8
mfeat-pixel	2000	240	0	240	10	4-6
mol-splice-junction	3190	60	0	60	3	4-5
nursery	12960	8	0	8	4	2-4
optdigits	5620	64	64	0	10	-
page-blocks	5473	10	10	0	5	-
parkinsons	195	22	22	0	2	-
pendigits	10992	16	16	0	10	-
postoperative-patient	90	8	8	0	3	2-4
primary-tumor	339	17	0	17	21	2-3
qsar-biodegradation	1055	41	41	0	2	-
qualitative-bankruptcy	250	6	0	6	2	3
saheart	462	9	8	1	2	2
segment	2310	19	16	0	7	-
seismic-bumps	2584	18	14	4	2	2-3
sick	3772	29	7	22	2	2
solar-flare2	1066	12	0	6	3	2-8
sonar	208	60	60	0	2	-
soybean	683	35	0	35	19	2-7
spambase	4601	57	57	0	2	-
spect	267	22	0	22	2	2
spectf	349	44	44	0	2	-
spectrometer	531	101	100	1	48	4
splice	3190	60	0	60	3	4-6
sponge	76	44	0	44	3	2-9
tae	151	5	3	2	3	2
thoracic-surgery	470	16	3	13	2	2-7
tic-tac-toe	958	9	0	9	2	3
turkiye-student	5820	32	32	0	13	-
vehicle	946	18	18	0	4	-
vote	435	16	0	16	2	2
vowel	990	11	10	1	11	2
waveform	5000	40	40	0	3	-
wine	178	13	13	0	3	-
zoo	101	16	1	16	7	2

Table 9.3: Friedman ranks about Accuracy of mNB with different values of m for each noise level.

m	0% Noise	10% Noise	20% Noise	30% Noise
1	**7.93**	9.63	11.75	12.49
2	8.14	**8.95**	10.74	11.92
3	7.98	8.99	10.41	11.51
4	8.11	9.35	10.30	11.07
5	8.65	9.59	10.21	11.04
6	9.05	9.67	10.19	10.66
7	9.39	9.39	10.71	10.11
8	9.23	9.28	**9.99**	10.27
9	9.55	9.68	10.15	10.29
10	10.25	10.26	10.15	10.26
11	10.49	10.71	10.05	10.48
12	10.87	11.11	10.20	10.23
13	11.67	11.59	10.17	**9.52**
14	11.95	11.58	10.17	9.73
15	12.15	11.81	10.18	9.77
16	12.01	11.44	10.64	9.81
17	12.65	11.49	11.07	9.77
18	12.93	11.74	11.03	9.88
19	13.35	11.94	10.81	10.45
20	13.63	11.80	11.09	10.75

Table 9.4: Average Accuracy results of the NB models with different levels of added noise.

Algorithm	noise 0%	noise 10%	noise 20%	noise 30%
Classical NB	77.05	74.69	73.08	71.08
Laplace NB	77.33	75.44	74.09	72.19
mNB $_{\mathrm{BEST}m}$	*79.60*	*77.74*	*76.11*	*73.54*
ImNB	**79.88**	**78.67**	**77.17**	**74.64**

Table 9.5: Friedman ranks about the Accuracy of the NB algorithms with different percentages of added noise.

Algorithm	noise 0%	noise 10%	noise 20%	noise 30%
Classical NB	2.76	3.12	3.05	2.97
Laplace NB	2.97	2.81	2.77	2.66
mNB$_{\mathrm{BEST}m}$	*2.21*	*2.35*	*2.37*	*2.36*
ImNB	**2.07**	**1.73**	**1.81**	**2.01**

Table 9.6: Average values of the ELA measure obtained by the NB algorithms for each noise level.

Algorithm	noise 10%	noise 20%	noise 30%
Classical NB	0.3285	0.3494	0.3753
Laplace NB	0.3176	0.3351	0.3596
mNB$_{\mathrm{BEST}m}$	*0.2796*	*0.3001*	*0.3324*
ImNB	**0.2670**	**0.2858**	**0.3175**

9.4 Conclusions

Naïve Bayes (NB) is a fast and simple approach to classification that has obtained good results in practice, comparable with more sophisticated classification algorithms. The estimation of the probabilities is the key point of NB. Many years ago, Cestnik [47] proposed a new NB model with a new way of estimating the probabilities. Such a model takes the prior probabilities into account through a parameter m for the estimation of the conditional probabilities. We believe that the Cestnik approach has not received sufficient attention from the scientific community, and we think that this is why we have not found an extensive comparison where this model evidences its worth.

In this chapter, we have proposed a new NB model, called the Imprecise m-probability estimation Naïve Bayes (ImNB), that also takes the prior probabilities of the class values into account for the estimation of the conditional probabilities. Nonetheless, our proposal uses the well-established uncertainty measure on credal sets for the estimation of the prior probabilities, whereas the Cestnik model employs relative frequencies with Laplacian correction to estimate such probabilities. Therefore, it can be stated that our proposed ImNB is more robust to class noise than the Cestnik model.

An exhaustive experimental analysis has been carried out in this chapter with many datasets and several levels of added noise to compare the performance of our proposed ImNB, the Cestnik model, and NB with classical and Laplace estimations. Such an experimental analysis has highlighted the following points:

- The best choice of the parameter m in the Cestnik model depends on the level of noise in the data, being generally higher as there is more noise.
- ImNB and the Cestnik model with the tuned value of m perform far better than NB estimating the probabilities in a classical way and with Laplace smoothing, regardless of the level of noise in the data.
- Our proposed ImNB using its default m value always achieves better results than the Cestnik model with parameter tuning, although the differences are not statistically significant in some cases. ImNB always significantly outperforms NB with and without Laplace smoothing.

To summarize, in this chapter, it has been shown, with much more exhaustive experimentation than in [47], that the Cestnik model supposes a very considerable improvement over Laplace and classical estimations of the probabilities in NB. Furthermore, we have presented a new way of estimating probabilities in NB based on the m-probability estimation and imprecise probabilities that outperforms the Cestnik model.

10 IMPROVEMENTS OVER IMPRECISE CLASSIFICATION ALGORITHMS

10.1 Introduction

Classifiers sometimes predict a set of values of the class variable because the available information is not sufficient to point out a single class value. This type of prediction is known as imprecise prediction, and classifiers that make imprecise predictions are called imprecise classifies. The first method proposed for Imprecise Classification was the Naïve Credal Classifier (NCC) [62, 227]. It combines the IDM with the naïve assumption to make imprecise predictions. Afterwards, the first Imprecise Classification method based on a single Decision Tree, called Imprecise Credal Decision Tree (ICDT), was proposed by Abellán and Masegosa [10]. For building the tree, ICDT employs uncertainty measures on credal sets and, to classify instances at leaf nodes, it uses the well-established dominance criterion on the probability intervals at such leaf nodes. Abellán and Masegosa [10] experimentally showed that ICDT significantly outperforms NCC as the former method is far more informative than the latter.

ICDT uses the IDM for the uncertainty measures and the probability intervals at leaf nodes. As commented previously, the A-NPI-M is an imprecise probability model also based on reachable probability intervals that, unlike the IDM, does not assume previous knowledge about the data via a parameter. In precise classification, the A-NPI-M performs equivalently to the IDM with the best selection of the parameter [6]. For these reasons, in this chapter, we propose a new Imprecise Credal Decision Tree that utilizes the A-NPI-M for the uncertainty measures in the split criterion and for the probability intervals at leaf nodes (ICDT-ANPI). We carry out an experimental study to compare the proposed ICDT-ANPI with the existing ICDT using different values of the IDM parameter. The obtained results are consistent with the ones achieved in precise classification: the performance of ICDT is strongly influenced by the choice of the IDM parameter and ICDT-ANPI has equivalent performance to ICDT with the best choice of the parameter.

Moreover, in this chapter, we propose a new version of the NCC algorithm called the Extreme Prior Naive Credal Classifier (EP-NCC). For the estimation of the lower and upper conditional probabilities of the class values, unlike

NCC, EP-NCC takes the lower and upper prior probabilities into account. It is based on the idea of the NB for precise classification proposed by Cestnik, described in Section 4.4.1. We show that, with our proposed EP-NCC, the non-dominated states set is often smaller than with the existing NCC. In this way, EP-NCC solves the drawback found in [10] about the NCC method: it is not very informative as it predicts too many class values. Since EP-NCC predicts fewer class values than NCC, the risk of making erroneous predictions might be higher with the former method. Nevertheless, EP-NCC considers the extreme prior probabilities of the class variable to reduce the number of predicted class values and, thus, the risk of incorrect predictions may not be much higher than with NCC. We carry out exhaustive experimentation to compare the performance of NCC, EP-NCC, and ICDT. Such experimentation reveals that our proposed EP-NCC significantly outperforms the existing NCC and that EP-NCC obtains statistically equivalent results to the ones achieved by ICDT. Specifically, EP-NCC obtains significantly better results than NCC in the metrics corresponding to the number of non-dominated states; the performance of both algorithms in the metrics associated with making correct predictions is equivalent; even though EP-NCC is not as accurate as ICDT, the former method is more informative than the latter as it predicts fewer values of the class variable. The experimental analysis also shows that the processing time for EP-NCC is considerably lower than for ICDT.

It should be noted that there is no algorithm for Imprecise Classification so far that makes an ensemble of classifiers even though ensembles tend to improve the performance of individual classifiers. It might be because, as the predictions made by imprecise classifiers usually consist of a set of class values, it is not a trivial question to combine them, and there is no technique so far for this purpose. If the imprecise predictions are not combined properly, then it is quite probable that the performance of the ensemble is not better than the performance of a single imprecise classifier because an excessive reduction of the information can be produced. Hence, a technique for combining multiple imprecise predictions must achieve a good trade-off between *risk* (the real class value does not belong to the set of predicted ones) and *informativeness* (how many class values are predicted)

In this chapter, as a novelty, we propose an ensemble method for Imprecise Classification that combines the predictions made by the individual classifiers in such a way that the ensemble method is as informative as possible although it implies a higher risk of erroneous predictions. When many imprecise predictions are combined, there is a risk of loss of information. We will see that this does not happen with our proposed combination technique. Specifically, our proposed ensemble method consists of a Bagging scheme using the ICDT algorithm as the base classifier. The reasons are that Bagging has obtained good performance in precise classification and Decision Trees are appropriate to be used in ensembles because they encourage diversity, and the key issue for the success of an ensemble scheme is that the individual classifiers are not only accurate but also diverse. An experimental analysis highlights that our proposed Bagging method for Imprecise Classification performs significantly better than ICDT.

All the above-mentioned algorithms assume that all classification errors have the same importance. However, in practical applications, different classification errors usually yield different costs. The ICDT algorithm was adapted for cost-sensitive classification by Abellán and Masegosa [10]. Such an adaptation uses the same tree-building process as the original ICDT method. At leaf nodes, it determines a risk interval for each value of the class variable considering the costs of errors. Then, it obtains the non-dominated states set via a strong dominance criterion on these risk intervals. Abellán and Masegosa [10] also adapted NCC for cost-sensitive classification. They experimentally showed that the adaptation of ICDT outperforms the adaptation of NCC since the former is more informative. The mentioned adaptations are the only algorithms proposed so far for cost-sensitive Imprecise Classification.

It is important to remark that the adaptation of ICDT for cost-sensitive classification proposed so far does not take the error costs into account in the procedure to build the tree; it considers, at each step of the procedure, that all instances have the same importance, regardless of their class values. This is obviously not optimal in scenarios where different classification errors lead to different costs. Hence, we consider, as in the Weighted Decision Tree algorithm for precise classification, exposed in Section 4.8.1, that the instances with a higher cost of error of the corresponding class value should have more weight than the instances for which the error cost of their class value is lower. In addition, the adaptation of ICDT for cost-sensitive classification proposed so far uses the IDM in the building process and for the probability intervals at leaf nodes. As pointed out before, the IDM assumes previous knowledge about the data via a parameter, and there is no way so far of associating the optimal value of such a parameter with each dataset. The A-NPI-M is a non-parametric approach that solves this shortcoming.

In this chapter, we propose a new cost-sensitive Imprecise Credal Decision Tree that employs the A-NPI-M and considers weights for the training instances depending on the error costs of their class values, similar to Weighted-DT. In this way, for computing the uncertainty measures in the split criterion and determining the probability intervals for the class values at leaf nodes, the instances with a higher cost of error of their class value have more importance.

We also show that the criterion used by our developed cost-sensitive Imprecise Credal Decision Tree for classifying instances may be more informative than the one of the existing cost-sensitive ICDT. An experimental analysis is carried out to compare the original cost-sensitive ICDT using the IDM and the A-NPI-M and our proposed cost-sensitive Imprecise Credal Decision Tree. Such an experimental analysis highlights that the A-NPI-M obtains equivalent results to the IDM with the recommended value of the parameter when both models are used in the existing cost-sensitive ICDT and that the new cost-sensitive Imprecise Credal Decision Tree significantly outperforms the existing one; even though our proposed method obtains higher misclassification costs than the existing cost-sensitive ICDT, it is often more informative and achieves a better trade-off between low cost of incorrect classifications and informative predictions.

The rest of this chapter is structured as follows: In Section 10.2, we present the Imprecise Credal Decision Tree based on the A-NPI-M. Section 10.3 describes our proposed Extreme Prior Naïve Credal Classifier. The Bagging method for Imprecise Classification is introduced in Section 10.4. Section 10.5 describes our proposed cost-sensitive Imprecise Credal Decision Tree. This chapter is concluded in Section 10.6.

Within this section, let C be the class variable and $\Omega_C = \{c_1, c_2, \dots, c_K\}$ its set of possible values. Let $\left\{X^1, X^2, \dots, X^d\right\}$ denote the set of predictive attributes.

10.2 Imprecise Credal Decision tree with A-NPI-M

The difference between our proposed Imprecise Credal Decision Tree with the A-NPI-M (ICDT-ANPI) and the existing ICDT is the mathematical model utilized for the split criterion and to compute the probability intervals at leaf nodes: whereas the existing ICDT uses the IDM, our proposed ICDT-ANPI employs the A-NPI-M.

Let $\mathcal{D}$ be the subset of the training set associated with a certain node. Let $N^{\mathcal{D}}$ denote the total number of instances in $\mathcal{D}$ and $n^{\mathcal{D}}(c_j)$ the number of instances in $\mathcal{D}$ for which $C = c_j, \quad \forall j = 1, 2, \dots, K$. We have the following set of A-NPI-M probability intervals on C corresponding to $\mathcal{D}$:

$$\begin{aligned} \mathcal{I}^{\mathcal{D}}_{ANPI}(C) = \Bigg\{ I^{\mathcal{D}}_{ANPI}(c_j) = \Bigg[\max\left(\frac{n^{\mathcal{D}}(c_j) - 1}{N^{\mathcal{D}}}, 0\right), \\ \min\left(\frac{n^{\mathcal{D}}(c_j) + 1}{N^{\mathcal{D}}}, 1\right)\Bigg], \quad j = 1, 2, \dots, K \Bigg\}. \end{aligned} \tag{10.1}$$

As we know, this set of probability intervals is always reachable and gives rise to the following credal set on C:

$$\mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C) = \left\{p \in \mathcal{P}(C) \mid p(c_j) \in I_{\text{ANPI}}(c_j), \quad \forall j = 1, 2, \ldots, K\right\}, \tag{10.2}$$

where $\mathcal{P}(C)$ denotes the set of all probability distributions on C.

Similarly to ICDT, the basis of the split criterion of our proposed ICDT-ANPI is the maximum entropy on this credal set:

$$S^*\left(\mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C)\right) = \max_{p \in \mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C)} S(p). \tag{10.3}$$

The split criterion of ICDT-ANPI is the uncertainty-based information gain taking as a reference the maximum entropy on the A-NPI-M credal set on C, defined in Equation (10.3). Such a split criterion, for an attribute X^i that takes values in $\left\{x^i_1, x^i_2, \ldots, x^i_{t_i}\right\}$, is defined in the following way:

$$IIG(C, X^i)^{\mathcal{D}}_{\text{ANPI}} = S^*\left(\mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C)\right) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) S^*\left(\mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C \mid X^i = x^i_{r_i})\right), \tag{10.4}$$

where $S^*\left(\mathcal{P}^{\mathcal{D}}_{\text{ANPI}}(C \mid X^i = x^i_{r_i}\right)$ is the maximum entropy on the A-NPI-M credal set on C on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}$ and $P^{\mathcal{D}}(X^i = x^i_{r_i})$ is the probability that $X^i = x^i_{r_i}$ on $\mathcal{D}$, estimated through relative frequencies:

$$P^{\mathcal{D}}(X^i = x^i_{r_i}) = \frac{n^{\mathcal{D}}(x^i_{r_i})}{N^{\mathcal{D}}},$$

$n^D(x^i_{r_i})$ being the number of instances in $\mathcal{D}$ such that $X^i = x^i_{r_i}, \quad \forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

In order to classify instances, ICDT-ANPI computes a probability interval for each class value at each leaf node using the A-NPI-M. Let $\mathcal{L}$ be a leaf node. Let $N^{\mathcal{L}}$ denote the number of instances in $\mathcal{L}$ and $n^{\mathcal{L}}(c_j)$ the number of instances in $\mathcal{L}$ that satisfy $C = c_j, \quad \forall j = 1, 2, \ldots, K$. We have the following set of A-NPI-M probability intervals on C on $\mathcal{L}$:

$$\begin{aligned}\mathcal{I}^{\mathcal{L}}_{\text{ANPIM}}(C) = \Bigg\{\Bigg[&\max\left(\frac{n^{\mathcal{L}}(c_j) - 1}{N^{\mathcal{L}}}, 0\right), \\ &\min\left(\frac{n^{\mathcal{L}}(c_j) + 1}{N^{\mathcal{L}}}, 1\right)\Bigg], \quad j = 1, 2, \ldots, K\Bigg\}.\end{aligned} \tag{10.5}$$

A dominance criterion has to be applied to these intervals for obtaining the non-dominated states set. Since A-NPI-M probability intervals are always reachable, in this case, the stochastic and credal dominance criteria are equivalent. For this reason, in order to obtain the non-dominated states set, ICDT-ANPI uses the stochastic dominance criterion, which is much easier to check. Consequently, in ICDT-ANPI, a class value c_j dominates another one c_k at $\mathcal{L}$ if, and only if,

$$\max\left(\frac{n^{\mathcal{L}}(c_j)-1}{N^{\mathcal{L}}},0\right) > \min\left(\frac{n^{\mathcal{L}}(c_k)+1}{N^{\mathcal{L}}},1\right) \Leftrightarrow$$

$$\frac{n^{\mathcal{L}}(c_j)-1}{N^{\mathcal{L}}} > \frac{n^{\mathcal{L}}(c_k)+1}{N^{\mathcal{L}}} \Leftrightarrow n^{\mathcal{L}}(c_j)-1 > n^{\mathcal{L}}(c_k)+1.$$

Thereby, at the leaf node $\mathcal{L}$, the non-dominated states set predicted by ICDT-ANPI is determined as follows:

$$nds^{\mathcal{L}}_{\text{ICDT-ANPI}} = \left\{c_k \mid n^{\mathcal{L}}(c_k)+1 \geqslant n^{\mathcal{L}}(c_j)-1 \quad \forall j = 1,2,\ldots,K\right\}. \tag{10.6}$$

Similar to ICDT, for classifying an instance via ICDT-ANPI, a path from the root node to a leaf one is made by using the attribute values of that instance. Then, the stochastic dominance is applied to the probability intervals at that leaf node to obtain the non-dominated states set for the instance. Algorithm 17 summarizes the procedure to classify an instance with ICDT-ANPI.

Algorithm 17: Procedure to classify an instance with ICDT-ANPI.

```
Procedure Classify_ICDT-ANPI(ICDT-ANPI $\mathcal{T}$, instance with attribute
vector $\mathbf{x}$)
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the
   attribute vector $\mathbf{x}$.
2. Consider the set of A-NPI-M probability intervals on C at $\mathcal{L}$,
   $\mathcal{I}^{\mathcal{L}}_{ANPIM}(C)$, computed through Equation (10.5).
3. Obtain the non-dominated states set at $\mathcal{L}$ via the stochastic
   dominance criterion on $\mathcal{I}^{\mathcal{L}}_{ANPIM}(C)$:

         $h(\mathbf{x}) = nds^{\mathcal{L}}_{\text{ICDT-ANPI}}$,

   where $nds^{\mathcal{L}}_{\text{ICDT-ANPI}}$ is determined by means of Equation (10.6).
return $h(\mathbf{x})$
```

10.2.1 Experiments

For our experimental analysis, we base on the experimental study carried out by Abellán and Masegosa in [10], where ICDT and NCC were compared.

10.2.1.1 *Experimental settings*

- **Datasets**: In our experiments, 34 datasets have been used, which can be downloaded from the *UCI Machine Learning Repository*. The datasets have been chosen in such a way that they have at least three class values because, with only two values of the class variable, an imprecise classifier always predicts all class values or only one. Table 10.1 shows the most important characteristics of each dataset. We may note that the datasets are diverse concerning the number of instances, number of continuous and discrete features, number of values of the class variable, etc.

- **Preprocessing**: Missing values have been replaced with mean values for continuous features and with modal values for discrete attributes. Afterwards, the datasets have been discretized via Fayyad and Irani's discretization method.

- **Algorithms**: In this experimentation, we aim to compare the existing ICDT algorithm with three values of the IDM parameter, $s = 1$, $s = 2$, and $s = 3$, and our proposed ICDT-ANPI method.

- **Evaluation**: The main evaluation metrics employed to test the performance of the algorithms considered in this experimental study are DACC and MIC[1]. For understanding better the behavior of the classifiers, we also consider the average values of Determinacy, Single Accuracy, Indeterminacy Size, and Set Accuracy. All these metrics were detailed in Section 5.3.

- **Procedure**: For comparing the results of the classifiers, a 10-fold cross-validation procedure has been repeated 10 times for each dataset and algorithm. The same partitions have been used for all methods.

- **Software** and **Parameters**: The Weka software [218] has been employed for our experimental study. It has been started from the implementation available in this software for ICDT, and the necessary structures and methods for using ICDT-ANPI have been added. The Weka filters have been employed for the preprocessing. Also, for cross validation, the functionality available in Weka has been used.

  For ICDT, three values of the IDM parameter have been used: $s = 1$, $s = 2$ and $s = 3$. The rest of the parameters used in both algorithms

[1] Here, we use $\text{MIC}^{0/1}$ since both ICDT and ICDT-ANPI assume the same cost for all classification errors.

Table 10.1: Description of the datasets employed in our experiments for Imprecise Classification. Column "N" is the number of instances, column "Attr" is the number of attributes, column "Cont" is the number of continuous features, column "Disc" is the number of discrete features, column "K" is the number of class values, and column "Range" is the range of values of the discrete attributes.

Dataset	N	Attr	Cont	Disc	K	Range
anneal	898	38	6	32	6	2-10
arrhythmia	452	279	206	73	16	2
audiology	226	69	0	69	24	2-6
autos	205	25	15	10	7	2-22
balance-scale	625	4	4	0	3	-
car	1728	6	0	6	4	3-4
cmc	1473	9	2	7	3	2-4
dermatology	366	34	1	33	6	2-4
ecoli	366	7	7	0	7	-
flags	194	30	2	28	8	2-13
hypothyroid	3772	30	7	23	4	2-4
iris	150	4	4	0	3	-
letter	20000	16	16	0	26	-
lymphography	146	18	3	15	4	2-8
mfeat-pixel	2000	240	0	240	10	4-6
nursery	12960	8	0	8	4	2-4
optdigits	5620	64	64	0	10	-
page-blocks	5473	10	10	0	5	-
pendigits	10992	16	16	0	10	-
postop-patient-data	90	9	0	9	3	2-4
primary-tumor	339	17	0	17	21	2-3
segment	2310	19	16	0	7	-
soybean	683	35	0	35	19	2-7
spectrometer	531	101	100	1	48	4
splice	3190	60	0	60	3	4-6
sponge	76	44	0	44	3	2-9
tae	151	5	3	2	3	2
vehicle	946	18	18	0	4	-
vowel	990	11	10	1	11	2
waveform	5000	40	40	0	3	-
wine	178	13	13	0	3	-
zoo	101	16	1	16	7	2

have been the ones given by default in Weka. Let ICDT-IDMi denote ICDT-IDM with $s = i$, for $i = 1, 2, 3$.

- **Statistical evaluation**: In accordance with the recommendations given by Demšar [75] for statistical comparisons between the results obtained by three or more algorithms on many datasets, we have used the Friedman test to compare the performance of the classifiers considered here via DACC and MIC. If the null hypothesis of this test is rejected, then we compare the algorithms pairwise via the Nemenyi test. The level of significance utilized is $\alpha = 0.05$. We present the results of these tests by means of critical diagrams.

10.2.1.2 *Results and discussion*

Tables 10.2 and 10.3 show the average values and Friedman ranks corresponding to DACC and MIC, respectively. In both tables, the best results are marked in bold. The critical diagrams associated with DACC and MIC can be seen in Figures 10.1 and 10.2, respectively.

Table 10.2: Average values and Friedman ranks of ICDT and ICDT-ANPI for the DACC measure.

Algorithm	Average	Friedman Rank
ICDT-ANPI	0.7675	**1.9118**
ICDT-IDM1	**0.7763**	2.3382
ICDT-IDM2	0.7606	2.4853
ICDT-IDM3	0.7482	3.2647

Table 10.3: Average values and Friedman ranks of ICDT and ICDT-ANPI for the MIC measure.

Algorithm	Average	Friedman Rank
ICDT-ANPI	1.3414	**1.9706**
ICDT-IDM1	**1.3652**	2.4412
ICDT-IDM2	1.3334	2.5
ICDT-IDM3	1.3065	3.0882

The following points should be noted about these results:

- **Average values**: For both DACC and MIC, the highest average value is obtained by ICDT-IDM1, followed by ICDT-ANPI, ICDT-IDM2, and ICDT-IDM3.

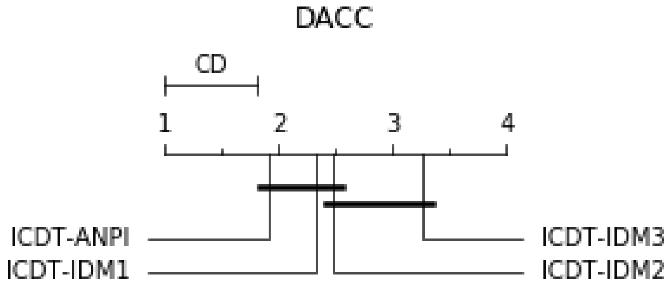

Figure 10.1: Critical diagram of ICDT and ICDT-ANPI for the DACC metric. CD = Critical Distance.

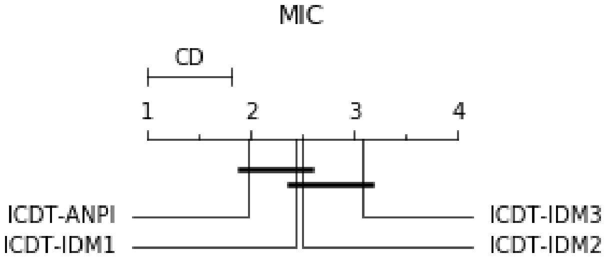

Figure 10.2: Critical diagram of ICDT and ICDT-ANPI for the MIC metric. CD = Critical Distance.

- **Friedman ranks**: The proposed ICDT-ANPI algorithm achieves the best Friedman rank for both DACC and MIC. Regarding ICDT, the higher is the value of the IDM parameter, the higher is the Friedman rank.
- **Nemenyi test**: In the critical diagram corresponding to DACC (Figure 10.1), it can be observed that the only algorithms that are not connected via a segment are ICDT-IDM1 and ICDT-IDM3 and ICDT-ANPI and ICDT-IDM3. In consequence, for DACC, both ICDT-ANPI and ICDT-IDM1 perform significantly better than ICDT-IDM3 according to the Nemenyi test. In the critical diagram associated with MIC (Figure 10.2), the only algorithms that are not connected via a segment are ICDT-ANPI and ICDT-IDM3. Thus, ICDT-ANPI significantly outperforms ICDT-IDM3 via the Nemenyi test for MIC. As ICDT-ANPI, ICDT-IDM1, and ICDT-IDM2 are connected via a segment in both critical diagrams, it can be stated that these three algorithms obtain statistically equivalent results according to the Nemenyi test for both DACC and MIC.

Hence, the performance of the ICDT algorithm depends on the choice of the s parameter. Regarding ICDT-ANPI, the results obtained by this algorithm are statistically equivalent to the ones obtained by ICDT with the best s parameter. Furthermore, ICDT-NPI performs significantly better than ICDT with the worst value of the s parameter.

Table 10.4 shows the average values of Determinacy, Single Accuracy, Set Accuracy and Indeterminacy size obtained by each algorithm.

Table 10.4: Average results obtained for basic metrics by of ICDT and ICDT-ANPI. Ind Size = Indeterminacy Size. Best scores are marked in bold.

Algorithm	Determinacy	Single Accuracy	Set Accuracy	Ind Size
ICDT-ANPI	0.9002	**0.8237**	**0.9561**	7.9381
ICDT-IDM1	**0.9477**	0.8023	0.8844	**5.2955**
ICDT-IDM2	0.8985	0.8119	0.9168	5.9313
ICDT-IDM3	0.8666	0.8151	0.9218	6.1346

We express the following comments about these results:

- **Determinacy**: ICDT-IDM1 achieves the highest average Determinacy value. It means that the highest number of instances precisely classified is obtained with ICDT-IDM1. ICDT-ANPI obtains the second-highest average Determinacy value, followed by ICDT-IDM2 and ICDT-IDM3.
- **Single Accuracy**: For the accuracy among the instances for which a single value of the class variable is predicted, ICDT obtains the worst per-

formance with $s = 1$. In the ICDT algorithm, the higher is the value of the s parameter, the better is the performance in Single Accuracy. The highest average Single Accuracy value is obtained by ICDT-ANPI.

- **Indeterminacy Size**: The lowest average Indeterminacy size value is achieved by ICDT-IDM1. Indeed, in ICDT, the lower is the s value, the lower is the average Indeterminacy size value. The highest average number of non-dominated states is obtained by ICDT-ANPI.

- **Set Accuracy**: The results achieved in Set Accuracy are similar to the ones obtained in Single Accuracy: ICDT performs better as the value of the s parameter is higher and ICDT-ANPI outperforms ICDT with the three s values considered.

Therefore, with ICDT-ANPI, the best trade-off between predicting only one class value and making correct predictions is attained. This algorithm obtains the second-highest score in Determinacy and the best score in Single Accuracy, whereas ICDT-IDM1, which achieves the highest average Determinacy value, obtains the worst results in Single Accuracy. Moreover, when there is more than one predicted value of the class variable, in the ICDT algorithm, the sizes of the non-dominated states sets are larger as the s value is higher and the largest sets are obtained with the ICDT-ANPI algorithm. Nonetheless, ICDT-ANPI obtains the highest proportion of instances for which the real class value is predicted and, in the ICDT method, this proportion is lower as the value of the s parameter is lower.

Summary of the results: The results obtained in this experimental analysis can be summarized in the following points:

- The ICDT algorithm predicts the real class value more frequently as the value of the s parameter is higher. However, if the s value is higher, then the predictions made by ICDT are less informative in the sense that the number of predicted class values sets is larger. This happens because the IDM is more imprecise as the value of the s parameter is higher.

- With ICDT-ANPI, although the non-dominated set is, on average, larger than with ICDT, it is achieved the best trade-off between predicting fewer values of the class variable and making correct predictions.

- The results obtained in the DACC and MIC measures allow us to deduce that ICDT-ANPI performs equivalently to ICDT with the best choice of the s parameter. In addition, the results obtained by ICDT-ANPI are significantly better than the ones obtained by ICDT with the worst s value.

10.3 Extreme Prior Naïve Credal Classifier

Within this section, we assume that the domain of each attribute X^i is finite, that is, the possible values of X^i are $\left\{x^i_1, x^i_2, \dots, x^i_{t_i}\right\}, \quad \forall i = 1, 2, \dots, d$.

We also assume that the lower probability of each value of the class variable is strictly greater than 0. Our proposed Extreme Prior Naïve Credal Classifier (EP-NCC) considers the Bayesian formula with the naïve assumption in the following way:

$$\begin{aligned}
&P\left(C = c_j \mid X^1 = x^1_{r_1}, X^2 = x^2_{r_2}, \dots, X^d = x^d_{r_d}\right) \sim \\
&P\left(C = c_j\right) \prod_{i=1}^{d} P\left(X^i = x^i_{r_i} \mid C = c_j\right) = P\left(C = c_j\right) \prod_{i=1}^{d} \frac{P\left(X^i = x^i_{r_i}, C = c_j\right)}{P\left(C = c_j\right)} = \\
&P\left(C = c_j\right) \prod_{i=1}^{d} \frac{P\left(X^i = x^i_{r_i}\right) P\left(C = c_j \mid X^i = x^i_{r_i}\right)}{P\left(C = c_j\right)} \sim \\
&P\left(C = c_j\right) \prod_{i=1}^{d} \frac{P\left(C = c_j \mid X^i = x^i_{r_i}\right)}{P\left(C = c_j\right)}, \\
&\forall j = 1, 2, \dots, K, \quad r_i = 1, 2 \dots, t_i, \quad i = 1, 2, \dots, d.
\end{aligned} \tag{10.7}$$

Let N_{tr} be the number of training instances and $n_{tr}(c_j)$ the number of training instances such that $C = c_j, \quad \forall j = 1, 2, \dots, K$. We have the set of IDM probability intervals on C, $\mathcal{I}_{IDM}(C)$, determined through Equation (5.11). We consider the IDM credal set on C consistent with such intervals:

$$\mathcal{P}_{EP}\left(C\right) = \left\{p \in \mathcal{P}(C) \mid p(c_j) \in IDM(c_j), \quad \forall j = 1, 2, \dots, K\right\}, \tag{10.8}$$

$\mathcal{P}(C)$ being the set of all probability distributions on C.

Let $n_{tr}\left(x^i_{r_i}\right)$ denote the number of training instances such that $X^i = x^i_{r_i}$ and $n_{tr}\left(x^i_{r_i, j}\right)$ the number of training instances that satisfy $X^i = x^i_{r_i} \wedge C = c_j, \quad \forall r_i = 1, 2, \dots, t_i, \quad i = 1, 2, \dots, d, \quad j = 1, 2, \dots, K$. For each $i = 1, 2, \dots, d$, and each $r_i = 1, 2, \dots, t_i$, we consider the following conditional credal set on the class variable:

$$\begin{aligned} \mathcal{P}_{EP}\left(C \mid x^i_{r_i}\right) = \Bigg\{ p \in \mathcal{P}(C) \mid \frac{n_{tr}\left(x^i_{r_i,j}\right) + s\underline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s} \\ \leqslant p(c_j) \leqslant \frac{n_{tr}\left(x^i_{r_i,j}\right) + s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s}, \quad \forall j = 1, 2, \dots, K \Bigg\}, \end{aligned} \tag{10.9}$$

where s is the IDM parameter and $\underline{p}_{EP}(c_k)$ and $\overline{p}_{EP}(c_k)$ are, respectively, the minimum and maximum values on c_j among all probability distributions belonging to $\mathcal{P}_{EP}(C)$. They are obtained as follows:

$$\underline{p}_{EP}(c_j) = \frac{n_{tr}(c_j)}{N_{tr} + s}, \quad \overline{p}_{EP}(c_j) = \frac{n_{tr}(c_j) + s}{N_{tr} + s}, \quad \forall j = 1, 2, \dots, K. \tag{10.10}$$

As NCC, we refer the credal sets $\mathcal{P}_{EP}(C)$ and $\mathcal{P}_{EP}\left(C \mid x^i_{r_i}\right)$ as local credal sets, $\forall r_i = 1, 2, \dots, t_i, \quad i = 1, 2, \dots, d$. The proposed EP-NCC algorithm considers, for each $i = 1, 2, \dots, d$, and each $r_i = 1, 2, \dots, t_i$, the set of joint probability distributions $\mathcal{P}_{EP}\left(C, x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}\right)$ resulting from making every possible combination of probability distributions on the local credal sets defined above:

$$\mathcal{P}_{EP}\left(C, x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}\right) = \left\{ p_c \prod_{i=1}^{d} \frac{p^i_{r_i}}{p_c}, \mid p_c \in \mathcal{P}_{EP}(C), \quad p_{r_i} \in \mathcal{P}_{EP}\left(C \mid x^i_{r_i}\right) \right\}. \tag{10.11}$$

Suppose now that it is wanted to classify an instance for which $X^i = x^i_{r_i}$, with $r_i \in \{1, 2, \dots, t_i\} \quad \forall i = 1, 2, \dots, d$.

The following result shows that the local credal sets $\mathcal{P}_{EP}\left(C \mid x^i_{r_i}\right)$ are defined by reachable probability intervals, $\forall r_i = 1, 2, \dots, t_i, \quad i = 1, 2, \dots, d$.

Proposition 10.3.1 *The set of probability intervals*

$$I_{C|x^i_{r_i}} = \left\{ \left[\frac{n_{tr}\left(x^i_{r_i,j}\right) + s\underline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s}, \frac{n_{tr}\left(x^i_{r_i,j}\right) + s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s} \right], \quad j = 1, 2, \dots, K \right\}$$

is reachable, $\forall r_i = 1, 2, \dots, t_i, \quad i = 1, 2, \dots, d$.

Proof:

For each $i = 1, 2, \dots, d$, it holds that:

$$\frac{n_{tr}\left(x^i_{r_i,j}\right) + s\underline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s} + \sum_{k=1,k\neq j}^{K} \frac{n_{tr}\left(x^i_{r_i,k}\right) + s\overline{p}_{EP}(c_k)}{n_{tr}\left(x^i_{r_i}\right) + s} =$$
$$\sum_{k=1}^{K} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i}\right) + s} + \frac{s}{n_{tr}\left(x^i_{r_i}\right) + s}\left(\underline{p}_{EP}(c_j) + \sum_{k=1,k\neq j}^{K} \overline{p}_{EP}(c_k)\right) \geqslant$$
$$\frac{n_{tr}\left(x^i_{r_i}\right)}{n_{tr}\left(x^i_{r_i}\right) + s} + \frac{s}{n_{tr}\left(x^i_{r_i}\right) + s} = 1, \quad \forall j = 1, 2, \dots, K.$$

since sets of IDM probability intervals are reachable.

Analogously, it can be checked that

$$\frac{n_{tr}\left(x^i_{r_i,j}\right) + s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s} + \sum_{k=1,k\neq j}^{K} \frac{n_{tr}\left(x^i_{r_i,k}\right) + s\underline{p}_{EP}(c_k)}{n_{tr}\left(x^i_{r_i}\right) + s} \leqslant 1, \quad \forall j = 1, 2, \dots, K,$$

and Proposition 2.2.6 allows us to conclude that $I_{C|x^i_{r_i}}$ is reachable.

□

Therefore, in our case, the credal and stochastic dominance criteria are equivalent. Consequently, we only need to analyze the relations among the bounds of the predicted intervals to obtain the non-dominated states set.

We consider

$$\begin{aligned}\underline{p}_{EP}(c_j \mid x^i_{r_i}) &= \min_{p_{ji}\in\mathcal{P}_{EP}\left(C|x^i_{r_i}\right)} \{p_{ji}(c_j \mid x^i_{r_i})\},\\ \overline{p}_{EP}(c_j \mid x^i_{r_i}) &= \max_{p_{ji}\in\mathcal{P}_{EP}\left(C|x^i_{r_i}\right)} \{p_{ji}(c_j \mid x^i_{r_i})\}, \quad \forall j = 1, 2, \dots K, \quad i = 1, 2, \dots, d.\end{aligned} \tag{10.12}$$

Clearly,

$$\underline{p}_{EP}(c_j \mid x^i_{r_i}) = \frac{n_{tr}\left(x^i_{r_i,j}\right) + s\underline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s}, \quad \overline{p}_{EP}(c_j \mid x^i_{r_i}) = \frac{n_{tr}\left(x^i_{r_i,j}\right) + s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right) + s},$$

$\forall j = 1, 2, \dots K, \quad i = 1, 2, \dots, d.$

We may observe that:

$$
\begin{aligned}
&\min_{p_c \in \mathcal{P}_{EP}(C), p^i_{r_i} \in \mathcal{P}_{EP}\left(C|x^i_{r_i}\right)} \left\{ p_c(c_j) \prod_{i=1}^{d} \frac{p^i_{r_i}(c_j \mid x^i_{r_i})}{p_c(c_j)} \right\} = \\
&\overline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{\underline{p}_{EP}(c_j \mid x^i_{r_i})}{\overline{p}_{EP}(c_j)} = \overline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\underline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right)+s}}{\overline{p}_{EP}(c_j)}, \\
&\max_{p_c \in \mathcal{P}_{EP}(C), p^i_{r_i} \in \mathcal{P}_{EP}\left(C|x^i_{r_i}\right)} \left\{ p_c(c_j) \prod_{i=1}^{d} \frac{p^i_{r_i}(c_j \mid x^i_{r_i})}{p_c(c_j)} \right\} = \\
&\underline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{\overline{p}_{EP}(c_j \mid x^i_{r_i})}{\underline{p}_{EP}(c_j)} = \underline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right)+s}}{\underline{p}_{EP}(c_j)}, \quad \forall j = 1, 2, \ldots K.
\end{aligned}
$$

Thus, according to EP-NCC, a class value c_k dominates another one c_j if, and only if:

$$
\begin{aligned}
&\overline{p}_{EP}(c_k) \prod_{i=1}^{d} \frac{\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\underline{p}_{EP}(c_k)}{n_{tr}\left(x^i_{r_i}\right)+s}}{\overline{p}_{EP}(c_k)} \geqslant \underline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\overline{p}_{EP}(c_j)}{n_{tr}\left(x^i_{r_i}\right)+s}}{\underline{p}_{EP}(c_j)} \Leftrightarrow \\
&\overline{p}_{EP}(c_k) \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)+s\underline{p}_{EP}(c_k)}{\overline{p}_{EP}(c_k)} \geqslant \underline{p}_{EP}(c_j) \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,j}\right)+s\overline{p}_{EP}(c_j)}{\underline{p}_{EP}(c_j)}, \\
&\forall j, k \in \{1, 2, \ldots, K\}.
\end{aligned}
$$

Since we are considering the credal set associated with the IDM for C, it holds that c_k dominates c_j under EP-NCC if, and only if:

$$\frac{n_{tr}\left(c_k\right)+s}{N_{tr}+s}\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{\frac{n_{tr}(c_k)+s}{N_{tr}+s}}\geqslant$$
$$\frac{n_{tr}\left(c_j\right)}{N_{tr}+s}\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)}{\frac{n_{tr}(c_j)}{N_{tr}+s}}\Leftrightarrow$$
$$\left(n_{tr}\left(c_k\right)+s\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(c_k\right)+s}\geqslant$$
$$n_{tr}\left(c_j\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)}{n_{tr}\left(c_j\right)},\quad\forall j,k\in\{1,2,\ldots,K\}.$$

To sum up, with EP-NCC, when it is required to classify a new instance with attribute vector $\mathbf{x}=\left(x^1_{r_1},x^2_{r_2},\ldots,x^d_{r_d}\right)$, with $r_i\in\{1,2,\ldots,t_i\}\quad\forall i=1,2,\ldots,d$, the predicted non-dominated states set is determined by:

$$h^{EP-NCC}(\mathbf{x})=\left\{c_j\mid n_{tr}\left(c_j\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)}{n_{tr}\left(c_j\right)}>\right.$$
$$\left.\left(n_{tr}\left(c_k\right)+s\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(c_k\right)+s},\quad\forall k\in\{1,2,\ldots,K\}\setminus\{j\}\right\}.\tag{10.13}$$

10.3.1 Justification of the new Naïve Credal Classifier

The main difference between the existing NCC and our proposed EP-NCC is the determination of the non-dominated states set for a given instance.

Unlike NCC, EP-NCC narrows the probability interval estimated for each class value by taking the lower and upper prior probabilities into account for the estimation of the conditioned probability interval. Hence, intuitively, the probability intervals predicted by EP-NCC might be more informative than the probability intervals predicted by NCC.

Suppose that, for an instance that is wanted to be classified, the frequency of a class value is much higher than the frequency of another class value, the proportion being greater than the proportions of the conditioned class frequencies for all attribute values. Let us also assume that all conditional class frequencies of the latter class value are greater than zero. As we show below, in these situations, if the former class value dominates the latter class value under NCC, then the same happens under EP-NCC.

Proposition 10.3.2 *Suppose that it is wanted to classify an instance for which* $X^i = x^i_{r_i}$*, with* $r_i \in \{1, 2, \dots, t_i\}$ $\forall i = 1, 2, \dots, d$*. Let* c_k *and* c_j *be two class values, where* $k, j \in \{1, 2, \dots, K\}$*. Suppose that* $n_{tr}\left(x^i_{r_i,j}\right) > 0$ *and* $\frac{n_{tr}(c_k)}{n_{tr}(c_j)+s} \geqslant \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)}$*,* $\forall i = 1, 2, \dots, d$*. In this case, if* c_k *dominates* c_j *under NCC, then* c_k *dominates* c_j *under EP-NCC.*

Proof: Under our hypothesis of dominance under NCC:

$$n_{tr}(c_k) \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}(c_k) + s} \geqslant (n_{tr}(c_j) + s) \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,j}\right) + s}{n_{tr}(c_j) + s} \Rightarrow$$

$$\left(\frac{n_{tr}(c_k)}{n_{tr}(c_k) + s}\right) \left(\frac{n_{tr}(c_j) + s}{n_{tr}(c_k) + s}\right)^{d-1} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s} \geqslant 1. \tag{10.14}$$

In addition, by hypothesis, $n_{tr}\left(x^i_{r_i,j}\right) > 0$ and $\frac{n_{tr}(c_k)}{n_{tr}(c_j)+s} \geqslant \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)}$, $\forall i = 1, 2, \dots, d$. So, it holds that:

$$\begin{aligned}
& n_{tr}(c_k) n_{tr}\left(x^i_{r_i,j}\right) \geqslant (n_{tr}(c_j) + s) n_{tr}\left(x^i_{r_i,k}\right) \Rightarrow \\
& s\left(\frac{n_{tr}(c_k)}{N_{tr} + s}\right) n_{tr}\left(x^i_{r_i,j}\right) \geqslant s\left(\frac{n_{tr}(c_j) + s}{N_{tr} + s}\right) n_{tr}\left(x^i_{r_i,k}\right) \Rightarrow \\
& n_{tr}\left(x^i_{r_i,j}\right) n_{tr}\left(x^i_{r_i,k}\right) + s\left(\frac{n_{tr}(c_k)}{N_{tr} + s}\right) n_{tr}\left(x^i_{r_i,j}\right) \geqslant \\
& n_{tr}\left(x^i_{r_i,j}\right) n_{tr}\left(x^i_{r_i,k}\right) + s\left(\frac{n_{tr}(c_j) + s}{N_{tr} + s}\right) n_{tr}\left(x^i_{r_i,k}\right) \Rightarrow \\
& \frac{n_{tr}\left(x^i_{r_i,k}\right) + s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)} \geqslant \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)}, \quad \forall i = 1, 2, \dots, d.
\end{aligned}$$

Thereby:

$$\prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right) + s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)} \geqslant \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)}. \tag{10.15}$$

Now, since $n_{tr}\left(x^i_{r_i,j}\right) \leqslant n_{tr}\left(c_j\right) \quad \forall i = 1, 2, \ldots, d$:

$$\prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,j}\right) + s}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant \prod_{i=1}^{d} \frac{n_{tr}\left(c_j\right) + s}{n_{tr}\left(c_j\right)} = \left(\frac{n_{tr}(c_j) + s}{n_{tr}(c_j)}\right)^{d} \geqslant \left(\frac{n_{tr}(c_j) + s}{n_{tr}(c_j)}\right)^{d-1}.$$

In this way:

$$\prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,j}\right) + s}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant \left(\frac{n_{tr}(c_j) + s}{n_{tr}(c_j)}\right)^{d-1} \Rightarrow$$

$$(n_{tr}(c_j))^{d-1} \prod_{i=1}^{d} \frac{1}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant (n_{tr}(c_j) + s)^{d-1} \prod_{i=1}^{d} \frac{1}{n_{tr}\left(x^i_{r_i,j}\right) + s} \Rightarrow$$

$$\frac{(n_{tr}(c_j))^{d-1}}{(n_{tr}(c_k) + s)^{d-1}} \prod_{i=1}^{d} \frac{1}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant \frac{(n_{tr}(c_j) + s)^{d-1}}{(n_{tr}(c_k) + s)^{d-1}} \prod_{i=1}^{d} \frac{1}{n_{tr}\left(x^i_{r_i,j}\right) + s} \Rightarrow$$

$$\frac{(n_{tr}(c_j))^{d-1}}{(n_{tr}(c_k) + s)^{d-1}} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant \frac{(n_{tr}(c_j) + s)^{d-1}}{(n_{tr}(c_k) + s)^{d-1}} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s}.$$

In consequence:

$$\left(\frac{n_{tr}\left(c_j\right)}{n_{tr}\left(c_k\right) + s}\right)^{d-1} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant \left(\frac{n_{tr}\left(c_j\right) + s}{n_{tr}\left(c_k\right) + s}\right)^{d-1} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s}. \tag{10.16}$$

Hence, we obtain:

$$\left(\frac{n_{tr}\left(c_j\right)}{n_{tr}\left(c_k\right) + s}\right)^{d-1} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right) + s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(x^i_{r_i,j}\right) + s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)} \geqslant$$

Equation (10.15)

$$\left(\frac{n_{tr}\left(c_j\right)}{n_{tr}\left(c_k\right) + s}\right)^{d-1} \prod_{i=1}^{d} \frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)} \geqslant$$

Equation (10.16)

$$\left(\frac{n_{tr}\left(c_j\right)+s}{n_{tr}\left(c_k\right)+s}\right)^{d-1}\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)+s}\geqslant$$

$$\left(\frac{n_{tr}\left(c_k\right)}{n_{tr}\left(c_k\right)+s}\right)\left(\frac{n_{tr}\left(c_j\right)+s}{n_{tr}\left(c_k\right)+s}\right)^{d-1}\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)}{n_{tr}\left(x^i_{r_i,j}\right)+s}\geqslant 1,$$

where the last inequality is due to Equation (10.14).

Therefore:

$$\left(\frac{n_{tr}\left(c_j\right)}{n_{tr}\left(c_k\right)+s}\right)^{d-1}\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)}\geqslant 1\Rightarrow$$

$$\left(\frac{1}{n_{tr}\left(c_k\right)+s}\right)^{d-1}\prod_{i=1}^{d}\left(n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}\left(c_k\right)}{N_{tr}+s}\right)\right)\geqslant$$

$$\left(\frac{1}{n_{tr}\left(c_j\right)}\right)^{d-1}\prod_{i=1}^{d}\left(n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}\left(c_j\right)+s}{N_{tr}+s}\right)\right)\Rightarrow$$

$$\left(n_{tr}\left(c_k\right)+s\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,k}\right)+s\left(\frac{n_{tr}(c_k)}{N_{tr}+s}\right)}{n_{tr}\left(c_k\right)+s}\geqslant n_{tr}\left(c_j\right)\prod_{i=1}^{d}\frac{n_{tr}\left(x^i_{r_i,j}\right)+s\left(\frac{n_{tr}(c_j)+s}{N_{tr}+s}\right)}{n_{tr}\left(c_j\right)},$$

which implies that c_k dominates c_j under EP-NCC.

□

The following example illustrates a case in which, intuitively, it makes much sense that a class value dominates another one, and such a dominance case is verified with EP-NCC but not with NCC.

Example 10.3.1 *Suppose that there are two attributes, namely X^1 and X^2. Let us assume that each attribute X^i takes values in $\left\{x^i_1, x^i_2\right\}$, for $i = 1,2$. Let C be the class variable, whose possible values are $\{c_1, c_2\}$. Suppose that $n_{tr}\left(c_1\right) = 48$ and $n_{tr}\left(c_2\right) = 2$. Let us assume the following arrangement of the class values for each one of the possible values of the attributes:*

$$\begin{aligned}X^1 = x^1_1 &\rightarrow n_{tr}\left(x^1_{1,1}\right) = 48, \quad n_{tr}\left(x^1_{1,2}\right) = 1;\\ X^1 = x^1_2 &\rightarrow n_{tr}\left(x^1_{2,1}\right) = 0, \quad n_{tr}\left(x^1_{2,2}\right) = 1;\\ X^2 = x^2_1 &\rightarrow n_{tr}\left(x^2_{1,1}\right) = 15, \quad n_{tr}\left(x^2_{1,2}\right) = 1;\\ X^2 = x^2_2 &\rightarrow n_{tr}\left(x^2_{2,1}\right) = 33, \quad n_{tr}\left(x^2_{2,2}\right) = 1.\end{aligned}$$

We assume the value $s = 1$ *for the IDM parameter. Suppose that it is required to classify an instance for which* $X^1 = x^1_2$ *and* $X^2 = x^2_2$*. Then:*

$$
\begin{aligned}
&(n_{tr}(c_1)+1) \times \frac{n_{tr}\left(x^1_{2,1}\right) + \left(\frac{n_{tr}(c_1)}{N_{tr}+1}\right)}{n_{tr}(c_1)+1} \times \frac{n_{tr}\left(x^2_{2,1}\right) + \left(\frac{n_{tr}(c_1)}{N_{tr}+1}\right)}{n_{tr}(c_1)+1} = \\
&49 \times \frac{\frac{48}{51}}{49} \times \frac{33+\frac{48}{51}}{49} = 0.6519 > 0.5605 = 2 \times \frac{1+\frac{3}{51}}{2} \times \frac{1+\frac{3}{51}}{2} = \\
&n_{tr}(c_2) \times \frac{n_{tr}\left(x^1_{2,2}\right) + \left(\frac{n_{tr}(c_2)+1}{N_{tr}+1}\right)}{n_{tr}(c_2)} \times \frac{n_{tr}\left(x^2_{2,2}\right) + \left(\frac{n_{tr}(c_2)+1}{N_{tr}+1}\right)}{n_{tr}(c_2)}.
\end{aligned}
$$

In consequence, c_1 *dominates* c_2 *under EP-NCC. However, it does not happen under NCC since:*

$$
\begin{aligned}
&n_{tr}(c_1) \times \frac{n_{tr}\left(x^1_{2,1}\right)}{n_{tr}(c_1)+1} \times \frac{n_{tr}\left(x^2_{2,1}\right)}{n_{tr}(c_1)+1} = 48 \times 0 \times \frac{33}{49} = 0 < \frac{4}{3} = \\
&3 \times \frac{2}{3} \times \frac{2}{3} = (n_{tr}(c_2)+1) \times \frac{n_{tr}\left(x^1_{2,2}\right)+1}{n_{tr}(c_2)+1} \times \frac{n_{tr}\left(x^2_{2,2}\right)+1}{n_{tr}(c_2)+1}.
\end{aligned}
$$

In this case, the prediction made by EP-NCC about the dominance of c_1 *on* c_2 *is intuitively more coherent than the one made by NCC.*

Therefore, the predictions of EP-NCC are probably more informative than the predictions of NCC. It is such an important issue for our proposal. In the previous example, we have observed a problematic situation of NCC: when only one lower conditional probability of a class value is equal to 0, the lower probability predicted by NCC for that value of the class variable is equal to 0, even though the rest of the lower conditional probabilities are very high. Thus, that class value does not dominate any other only due to that lower conditional probability, which is incoherent. This problem is solved with our proposed EP-NCC because it considers the lower prior probabilities of the class values for the estimation of the lower conditional probabilities.

Our proposed EP-NCC assumes more risk of making incorrect predictions than NCC since its predicted non-dominated states set is often smaller. However, this risk is controlled because the probability interval predicted for each class value is narrowed by taking the extreme prior probabilities into account. In addition, EP-NCC solves some problematic situations of NCC, such as the one found in Example 10.3.1.

To summarize, EP-NCC is more appropriate than NCC for Imprecise Classification as the former method is more informative than the latter without assuming a much higher risk of making erroneous predictions. This fact is corroborated in Section 10.3.2 with an exhaustive experimental analysis.

10.3.2 Experimental study

In this experimental analysis, we aim to compare the performance of the existing NCC, our proposed EP-NCC, and ICDT.

10.3.2.1 *Experimental setup*

For this experimentation, we take as a reference the experimental analysis carried out by Abellán and Masegosa in [10], where the NCC and the ICDT methods were compared.

- **Datasets**: The three algorithms considered in this experimentation have been applied to the 34 classification datasets that we used in the experiments of the previous section, were we checked the performance of ICDT-ANPI. The most important characteristics of each dataset can be seen in Table 10.1.

- **Preprocessing**: Missing values have been replaced with mean values for continuous attributes and modal values for discrete features. After that, continuous features have been discretized by following Fayyad and Irani's discretization method.

- **Algorithms**: Three algorithms have been used in this experimental study: NCC, EP-NCC, and ICDT.

- **Evaluation**: The main evaluation metrics used for checking the performance of the algorithms considered here are DACC and MIC[2]. For a deeper analysis of the behavior of the algorithms, we also consider Determinacy, Single Accuracy, Indeterminacy Size, and Set Accuracy. All these metrics were detailed in Section 5.3.

  Furthermore, we aim to compare the computational complexity of the algorithms considered in this experimentation. For this purpose, we consider the processing time, in milliseconds, of the algorithms.

- **Procedure**: In order to compare the performance of the classifiers considered in this experimental study, for each dataset and algorithm, a 10-fold cross-validation procedure has been repeated 10 times. The same partitions have been used for all algorithms.

[2] Here, we use $\text{MIC}^{0/1}$ because we use classifiers that assume the same cost for all classification errors.

- **Software** and **Parameters**: We have used the Weka software for our experiments. We have utilized the implementation available in Weka for ICDT, and we have added the necessary structures and methods for using NCC and EP-NCC. We have employed the Weka filters for the preprocessing. Also, for cross-validation, we have used the functionality available in Weka.

  We have employed the value $s = 1$ for the IDM parameter for the three algorithms as it is one of the recommended by Walley [209] and is the one used in the experimental analysis carried out in [10], where NCC and ICDT were compared. The rest of the parameters utilized for all algorithms have been the ones given by the default in Weka.

- **Statistical evaluation**: Following the recommendations of Demšar [75] for statistical comparisons between the results obtained by three or more methods on many datasets, we have used the Friedman test to compare the performance of NCC, EP-NCC, and ICDT via the evaluation metrics considered here. If the null hypothesis of this test is rejected, then we compare the algorithms pairwise through the Nemenyi test. The level of significance utilized is $\alpha = 0.05$. We present the results of these tests via critical diagrams.

10.3.2.2 *Results and discussion*

Tables 10.5 and 10.6 show that average values and Friedman ranks corresponding to DACC and MIC, respectively. In both tables, the best results are marked in bold fonts. The critical diagram corresponding to DACC (MIC) can be seen in Figure 10.3 (10.4).

Table 10.5: Average values and Friedman ranks of NCC, EP-NCC, and ICDT corresponding to DACC.

Algorithm	Average	Friedman rank
NCC	0.6237	2.6765
EP-NCC	**0.7810**	**1.5882**
ICDT	0.7763	1.7353

We must remark the following points about these results:

- Both ICDT and EP-NCC achieve a higher average value and a lower Friedman rank than NCC in MIC and DACC. Moreover, in the critical diagrams, NCC is not connected through a segment with the other two

Table 10.6: Average values and Friedman ranks of NCC, EP-NCC, and ICDT corresponding to MIC.

Algorithm	Average	Friedman rank
NCC	1.3042	2.8235
EP-NCC	**1.3529**	**1.5882**
ICDT	1.7676	1.8235

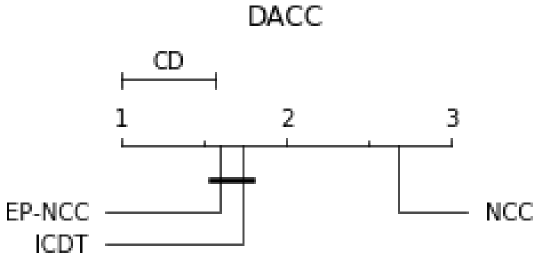

Figure 10.3: Critical diagram corresponding to NCC, EP-NCC, and ICDT for the DACC metric. CD = Critical Distance.

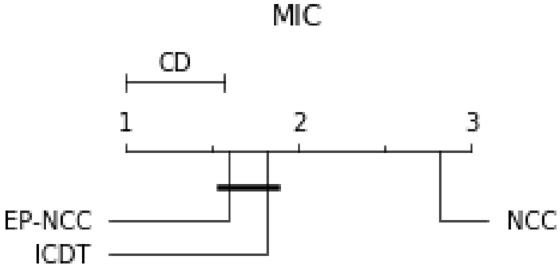

Figure 10.4: Critical diagram corresponding to NCC, EP-NCC, and ICDT for the MIC metric. CD = Critical Distance.

algorithms. Consequently, both ICDT and EP-NCC perform significantly better than NCC according to the Nemenyi test in MIC and DACC.

- In both DACC and MIC, the average value of EP-NCC is higher than the average value of ICDT, and the Friedman rank of EP-NCC is lower than the Friedman rank of ICDT. Nevertheless, in the critical diagrams associated with these metrics, these two algorithms are connected via a segment. Thereby, EP-NCC and ICDT perform equivalently according to the Nemenyi test in both DACC and MIC.

Tables 10.7 and 10.8 illustrate, respectively, the average values and average Friedman ranks obtained by each classifier in Determinacy, Single Accuracy, Indeterminacy size, and Set Accuracy. Again, the best results are marked in bold. The critical diagrams corresponding to these measures are shown in Figures 10.5, 10.6, 10.7, and 10.8.

Table 10.7: Average results obtained by NCC, EP-NCC, and ICDT in Determinacy, Single Accuracy, Indeterminacy size and Set Accuracy.

Metric	NCC	EP-NCC	ICDT
Determinacy	0.6037	0.9150	**0.9477**
Single Accuracy	**0.8692**	0.8165	0.8058
Indeterminacy size	4.4537	**2.0999**	5.3294
Set Accuracy	0.8698	0.8389	**0.8999**

Table 10.8: Average Friedman ranks obtained by NCC, EP-NCC, and ICDT in Determinacy, Single Accuracy, Indeterminacy size and Set Accuracy.

Metric	NCC	EP-NCC	ICDT
Determinacy	2.8235	**1.5**	1.6765
Single Accuracy	**1.3226**	2.3548	2.3226
Indeterminacy size	2.1875	**1.0469**	2.7656
Set Accuracy	1.9844	2.3594	**1.6562**

The following points should be noted about the results obtained in these measures:

- NCC obtains, by far, the worst performance in predicting a single class value due to the results obtained in **Determinacy**. Indeed, in this metric, NCC gets the highest Friedman rank and the lowest average value. Furthermore, in the critical diagram of Figure 10.5, NCC is not connected

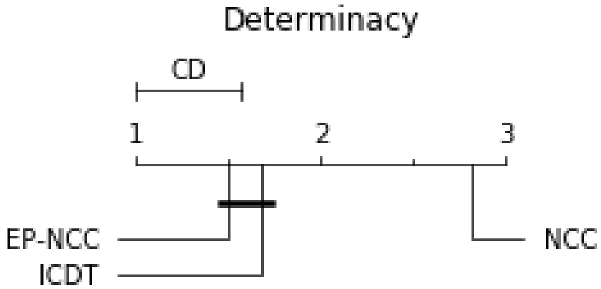

Figure 10.5: Critical diagram corresponding to NCC, EP-NCC, and ICDT for Determinacy. CD = Critical Distance.

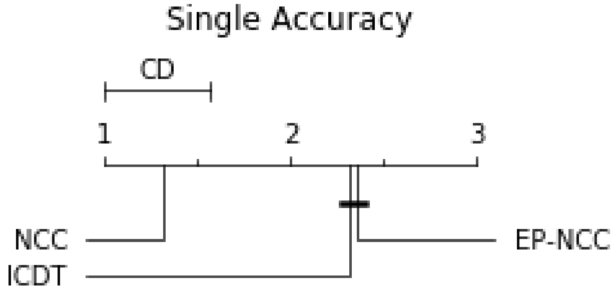

Figure 10.6: Critical diagram corresponding to NCC, EP-NCC, and ICDT for Single Accuracy. CD = Critical Distance.

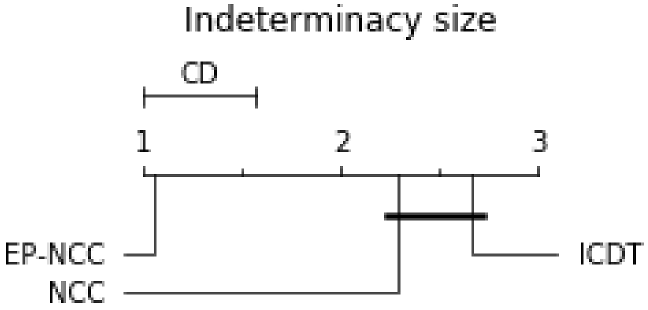

Figure 10.7: Critical diagram corresponding to NCC, EP-NCC, and ICDT for Indeterminacy size. CD = Critical Distance.

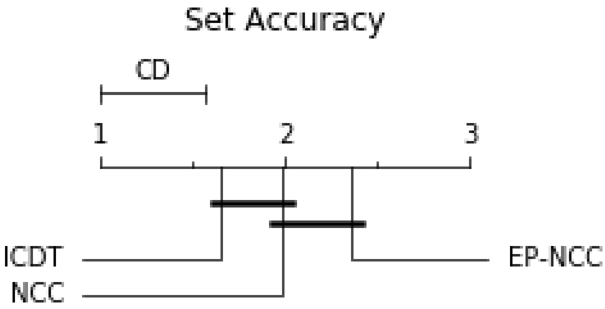

Figure 10.8: Critical diagram corresponding to NCC, EP-NCC, and ICDT for Set Accuracy. CD = Critical Distance.

via a segment with EP-NCC or ICDT, which implies that these two algorithms significantly outperform NCC via the Nemenyi test in Determinacy. In this metric, EP-NCC obtains a lower Friedman rank than ICDT, whereas the latter algorithm obtains a higher average value than the former. However, EP-NCC and ICDT are connected via a segment in the critical diagram of Figure 10.5. Hence, in Determinacy, ICDT and EP-NCC have equivalent performance according to the Nemenyi test.

- Regarding **Single Accuracy**, which measures the accuracy among the instances for which a single class value is predicted, NCC achieves, by far, the highest average value and the lowest Friedman rank. In addition, NCC is not connected by a segment with the other two algorithms in the critical diagram of Figure 10.6 and, thus, it significantly outperforms EP-NCC and ICDT via the Nemenyi test in Single Accuracy. Nevertheless, we should remark that, with NCC, the proportion of instances precisely classified is far lower than with the other two algorithms. EP-NCC and ICDT are connected through a segment in the critical diagram of Figure 10.6. So, there are no statistically significant differences via the Nemenyi test between these two methods in Single Accuracy.

- The results obtained in **Indeterminacy size** allow deducing that EP-NCC achieves, by far, the lowest average number of non-dominated class values. In fact, it attains the highest average value and the lowest Friedman rank in this metric. Moreover, EP-NCC is not connected through a segment with the other two algorithms in the corresponding critical diagram (Figure 10.7). Therefore, EP-NCC performs significantly better than the other two algorithms in Indeterminacy Size according to the Nemenyi test. Even though NCC attains a lower average value and a lower

Friedman rank than ICDT in Indeterminacy size, these two algorithms are connected via a segment in the critical diagram of Figure 10.7. In consequence, in this measure, NCC and ICDT obtain statistically equivalent results according to the Nemenyi test.

- Concerning **Set Accuracy**, which measures the proportion of correct predictions among the instances for which more than a class value is predicted, ICDT obtains the best results according to Friedman rank and average values. Furthermore, ICDT and EP-NCC are not connected with a segment in the critical diagram of Figure 10.8 and, consequently, the former algorithm significantly outperforms the latter via the Nemenyi test in Set Accuracy. Although NCC obtains a higher average value and a lower Friedman rank than ICDT in Set Accuracy, these two algorithms are connected via a segment in the critical diagram of Figure 10.8. Hence, there are no statistically significant differences according to the Nemenyi test between the results obtained by ICDT and NCC in Set Accuracy.

Table 10.9 shows the average values and Friedman ranks corresponding to the processing times of NCC, EP-NCC, and ICDT. Again, the best results are marked in bold. The critical diagram associated with the computational time can be seen in Figure 10.9.

Table 10.9: Average values and Friedman ranks obtained by NCC, EP-NCC, and ICDT corresponding to the processing time results.

Algorithm	Average	Friedman rank
NCC	0.0006	1.5588
EP-NCC	**0.0005**	**1.4412**
ICDT	0.0424	3

We may note that ICDT requires, by far, the highest computational time. It should be noted that it obtains the worst result in terms of processing time for all datasets since its average Friedman rank is equal to 3. In the critical diagram of Figure 10.9, ICDT is not connected via segments with the other two algorithms. Thereby, it performs significantly worse than NCC and EP-NCC according to the Nemenyi test in terms of computational time. In addition, the average processing time obtained by ICDT is much higher than the ones achieved by NCC and EP-NCC. NCC obtains a higher average value than EP-NCC, and the Friedman rank achieved by NCC is also higher. However, there are no statistically significant differences between these two algorithms in terms of processing time according to the Nemenyi test as they are connected with a segment in the critical diagram of Figure 10.9.

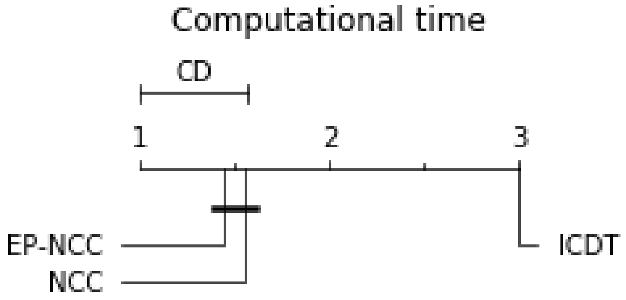

Figure 10.9: Critical diagram associated with NCC, EP-NCC, and ICDT for computational time. CD = Critical Distance.

Summary of the results: The results obtained in this experimental analysis can be summarized in the following points:

- EP-NCC outperforms NCC. Specifically, EP-NCC performs better than NCC in predicting fewer class values and, thus, EP-NCC is more informative than NCC, as we argued in Section 10.3.1. According to the results obtained in Single Accuracy and Set Accuracy, NCC performs slightly better than EP-NCC in making the right predictions. It is because the predictions made by EP-NCC are more precise than the ones made by NCC and, thus, the former method assumes more risk of making incorrect predictions than the latter. Nonetheless, this is not as important as the fact that EP-NCC is more informative since the risk is not far higher, as we commented in Section 10.3.1, and it is highlighted in the results obtained in DACC and MIC.

- ICDT and EP-NCC have equivalent performance. Both algorithms obtain statistically equivalent results in predicting only one value of class variable and in the accuracy among precise predictions. Regarding instances imprecisely classified, ICDT outperforms EP-NCC in making the right predictions, while EP-NCC performs significantly better than ICDT in predicting fewer values of the class variable.

- The processing times of NCC and EP-NCC are very similar. The ICDT algorithm requires a much higher computational time than NCC and EP-NCC.

10.4 Bagging of Imprecise Credal Decision Trees

In this section, a Bagging scheme for Imprecise Classification that uses the ICDT algorithm as base classifier is proposed. We call this new method the Bagging of Imprecise Credal Decision Trees (Bagging-ICDT).

For building the individual classifiers, Bagging-ICDT uses a similar idea to the Bagging scheme for precise classification. For each individual ICDT, a bootstrapped sample of the original training set is selected. Then, an Imprecise Classification model is learned using such a bootstrapped sample and our base classification algorithm, ICDT.

The key point of the proposed Bagging scheme for Imprecise Classification is how to combine the predictions made by the individual classifiers. Remark that, in precise classification, combining the predictions is as simple as taking the majority vote. However, in Imprecise Classification, it is not a trivial question since the individual classifiers may not return a unique value of the class variable, but they might predict a set of non-dominated class values. Actually, there are multiple ways of combining imprecise predictions because a class value might be predicted as dominated by some classifiers and as non-dominated by others. The crucial issue is how to determine, taking into account the number of classifiers that predict that a class value is dominated, the threshold to decide whether that class value is dominated for the final prediction. In fact, this consists of a trade-off between *risk* and *information*. Here, the term *risk* is used to denote the possibility of not including the real class value in the predicted non-dominated states set, and the term *information* indicates how precise the prediction is, i.e, how many class values are predicted as non-dominated. Logically, more information implies more risk. We consider that our proposed technique is closer to the risk because it predicts the class values with the minimum level of dominance.

If all the class values that are predicted as non-dominated by at least one classifier are finally predicted as non-dominated, then the probability of making an erroneous prediction is minimum. Nonetheless, in these situations, the predicted non-dominated states set may be composed of almost all values of the class variable. In this way, the predictions are probably hardly informative and, thus, the Bagging scheme might not be very useful. For this reason, our strategy consists of the opposite extreme: we want that the Bagging scheme is as informative as possible, even though this implies a higher risk of erroneous prediction.

Therefore, when an instance is wanted to be classified in our proposed Bagging-ICDT algorithm, for each class value, the number of classifiers that predict such a class value as dominated is counted. We call that number of votes the number of *votes against*. The non-dominated states set predicted by Bagging-ICDT is composed of those class values with the minimum number of votes against.

Algorithm 18 summarizes our proposed Bagging-ICDT method.

Algorithm 18: Bagging scheme with ICDT.

```
Procedure Bagging-ICDT (Training set $\mathcal{D}$, number of ICDTs $n\_trees$)
for $i = 1$ to $n\_trees$ do
    Select a bootstrapped sample with replacement, $\mathcal{D}_i$, from $\mathcal{D}$
      ($|\mathcal{D}_i| = |\mathcal{D}|$)
    Build a classifier $\mathcal{C}_i$ using the ICDT algorithm and $\mathcal{D}_i$ as training set
For classifying an instance with attribute vector $\mathbf{x}$
for $j = 1$ to $K$ do
    Let $va_j$ be the number of classifiers that predict $c_j$ as dominated for
      $\mathbf{x}$
$min\_against \leftarrow \min_{j=1,\cdots,K} va_j$
$h^{Bagg-ICDT}(\mathbf{x}) \leftarrow \left\{c_j \mid va_j = min\_against, \quad 1 \leqslant j \leqslant K\right\}$
return $h^{Bagg-ICDT}(\mathbf{x})$
```

In summary, with our proposed Bagging-ICDT, we try to improve the performance of an individual ICDT by increasing the diversity through multiple ICDTs built with different training sets. In order to classify new instances, the predictions made by the individual classifiers are combined for the proposed Bagging scheme to be as informative as possible.

10.4.1 Experimentation

10.4.1.1 *Description of the experiments*

For our experimental analysis, we take as a reference the one carried out by Abellán and Masegosa in [10], where the ICDT method was proposed.

- **Datasets**: In this experimental study, we have employed the 34 classification datasets that we used in the experiments of the previous sections with Imprecise Classification algorithms. Table 10.1 shows the most important characteristics of such datasets.

- **Preprocessing**: Missing values have been replaced with mean values for continuous features and modal values for discrete attributes. Then, continuous attributes have been discretized via Fayyad and Irani's discretization method.

- **Algorithms**: Two algorithms have been considered in our experiments: ICDT and Bagging-ICDT.

- **Evaluation**: In order to evaluate the performance of the algorithms considered here, we principally consider the evaluation metrics DACC and MIC[3]. For a deeper analysis of the behavior of the algorithms, we also consider the average values of Determinacy, Single Accuracy, Indeterminacy Size, and Set Accuracy. All these metrics were described in Section 5.3.

- **Procedure**: For comparing the performance of ICDT and Bagging-ICDT, for each dataset and algorithm, a 10-fold cross-validation procedure has been repeated 10 times. The same partitions have been utilized for ICDT and Bagging-ICDT.

- **Software** and **Parameters**: We have used the Weka software for this experimental study. We have started from the implementation available in Weka for ICDT, and we have added the necessary structures and methods for Bagging-ICDT. For the preprocessing, we have utilized the Weka filters. Also, we have employed the functionality available in Weka for cross-validation.

  For the IDM parameter, we have used the value $s = 1$ for both algorithms since it is one of the values recommended by Walley [209] and requires a low computational cost. For Bagging-ICDT, we have used 100 trees as it is an appropriate number of classifiers for a Bagging scheme [37]. The rest of the parameters employed for both algorithms have been the ones given by the default in Weka.

- **Statistical evaluation**: Consistently with the recommendations of Demšar [75] for statistical comparisons between the results obtained by two algorithms on many datasets, we have used the Wilcoxon test with a level of significance of $\alpha = 0.05$ to compare the performance of ICDT and Bagging-ICDT via DACC and MIC.

  In addition, the Corrected Paired t-test has been employed to compare the performance of ICDT and Bagging-ICDT in each dataset. It is a corrected version of the Paired t-test implemented in Weka. This test checks whether one algorithm performs better than the other, on average, across all the training and test sets extracted from a cross-validation procedure repeated several times on a dataset.

[3] Here, we use $\mathrm{MIC}^{0/1}$ because we are using classifiers that assume the same cost for all classification errors.

10.4.1.2 *Results and discussion*

Tables 10.10 and 10.11 present the results obtained by each algorithm in each dataset in the DACC and MIC measures, respectively. In both tables, for each dataset, the best result is marked in bold. Moreover, these tables show, for each dataset, which algorithm performs better according to the Corrected Paired t-test (in case the differences are significant).

A summary of the results for both DACC and MIC evaluation metrics can be seen in Table 10.12. In concrete, for DACC and MIC, Table 10.12 shows the average value, the result of the Wilcoxon test, and the number of datasets where ICDT performs significantly better than Bagging-ICDT according to the Corrected Paired t-test and vice-versa.

We express the following comments about these results:

- From Tables 10.10 and 10.11, it can be observed that, for both DACC and MIC, Bagging-ICDT outperforms ICDT in almost all datasets. In fact, for DACC, ICDT only performs better than Bagging-ICDT in one dataset, and both algorithms obtain the same result in two datasets. In the rest of the datasets, the performance is better for Bagging-ICDT according to this measure. Similarly, for MIC, Bagging-ICDT obtains a better result than ICDT in all datasets except four. In three of them, ICDT outperforms Bagging-ICDT and, in the other one, both algorithms obtain the same result.

- As can be seen in Table 10.12, Bagging-ICDT performs significantly better than ICDT for both DACC and MIC according to the Wilcoxon test. Also, the average DACC value obtained by Bagging-ICDT is much higher than the average DACC value obtained by ICDT. The same happens with the average values of MIC.

- Furthermore, according to the Corrected Paired t-test, the number of datasets where Bagging-ICDT obtains significantly better results than ICDT is 16 for DACC and 14 for MIC. In contrast, for none of the two metrics, ICDT performs significantly better than Bagging-ICDT in any dataset according to the Corrected Paired t-test.

In this way, it can be concluded that Bagging-ICDT performs much better than ICDT, the differences being pretty considerable.

Table 10.13 shows the average results of Determinacy, Single Accuracy, Set Accuracy, and Indeterminacy size obtained by ICDT and Bagging-ICDT. The best results are marked in bold. Figure 10.10 allows us to observe better the

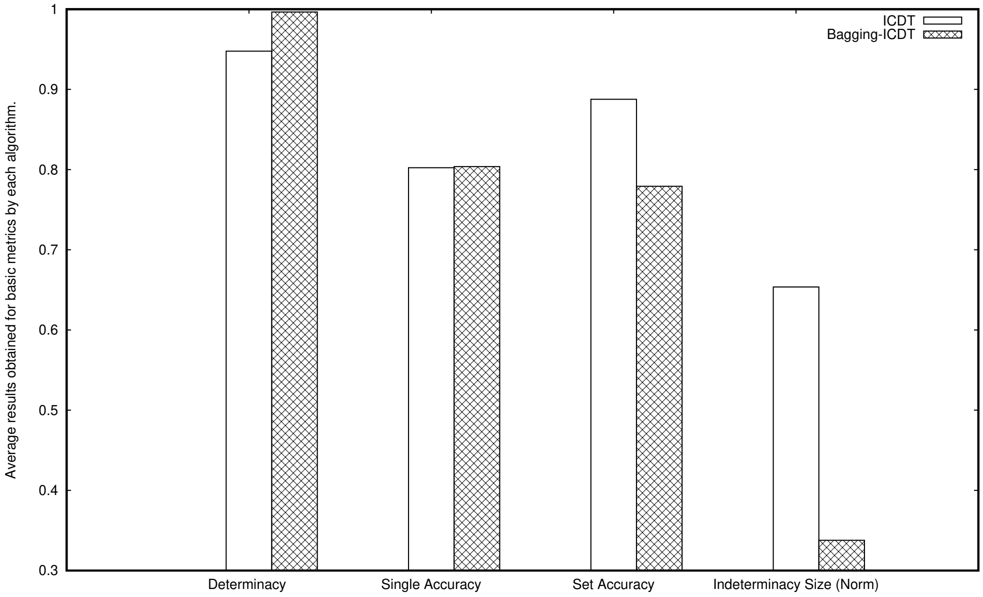

Figure 10.10: Graphic about the normalized average results obtained by ICDT and Bagging-ICDT in Determinacy, Single Accuracy, Indeterminacy size, and Set Accuracy.

differences between the results obtained by both algorithms in these metrics. The results shown in the graphic are normalized.

For each one of these metrics, the following points should be noted:

- **Determinacy**: The proportion of instances for which a single class value is predicted is higher for Bagging-ICDT.
- **Single Accuracy**: Between the instances precisely classified, the accuracy is similar for both algorithms
- **Indeterminacy size**: The imprecise predictions made by Bagging-ICDT are more informative than the imprecise predictions made by ICDT.
- **Set Accuracy**: Among the instances imprecisely classified, there are more erroneous predictions with Bagging-ICDT.

Due to the previous points, it can be stated that the predictions made by Bagging-ICDT are more informative than the ones made by ICDT, even though the error rate is a little bit higher with Bagging-ICDT.

Summary of the results: We can summarize the results obtained in this experimentation in the following issues:

- Bagging-ICDT is a far more informative classifier than ICDT, although the error rate is a little bit higher with the former algorithm.
- The results obtained in the main Imprecise Classification evaluation metrics proposed so far in the literature, DACC and MIC, allow concluding that our proposed Bagging-ICDT significantly outperforms ICDT.

Table 10.10: Complete results obtained by ICDT and Bagging-ICDT in DACC. In each row, ○ means that Bagging-ICDT significantly outperforms ICDT via the Corrected Paired t-test in the corresponding dataset; • indicates that ICDT performs significantly better than Bagging-ICDT according to the Corrected Paired t-test in the dataset of the row.

Dataset	ICDT	Bagging-ICDT	
anneal	0.9957	**0.9967**	
arrhythmia	0.6625	**0.7150**	○
audiology	0.7887	**0.8232**	
autos	0.7817	**0.8278**	○
balance-scale	0.6961	**0.6977**	
bridges-version1	0.6375	**0.6503**	
bridges-version2	0.5729	**0.6199**	
car	0.9168	**0.9299**	○
cmc	0.4884	**0.4931**	
dermatology	0.9405	**0.9500**	
ecoli	0.7993	**0.8054**	
flags	0.5554	**0.6034**	○
hypothyroid	0.9935	0.9935	
iris	0.9337	**0.9390**	
letter	0.7714	**0.8277**	○
lymphography	0.7275	**0.7591**	
mfeat-pixel	0.7702	**0.8837**	○
nursery	0.9628	**0.9654**	○
optdigits	0.7716	**0.8647**	○
page-blocks	0.9619	**0.9663**	○
pendigits	0.8812	**0.9175**	○
postoperative-patient-data	**0.7104**	0.7100	
primary-tumor	0.3815	**0.4239**	○
segment	0.9406	**0.9502**	○
soybean	0.9178	**0.9276**	
spectrometer	0.4430	**0.5127**	○
splice	0.9270	**0.9447**	○
sponge	0.9293	**0.9475**	
tae	0.4678	0.4678	
vehicle	0.6899	**0.7025**	
vowel	0.7635	**0.7953**	○
waveform	0.7371	**0.7777**	○
wine	0.9194	**0.9290**	
zoo	0.9592	**0.9612**	
Average	0.7763	**0.8023**	

Table 10.11: Complete results obtained by ICDT and Bagging-ICDT in MIC. In each row, ○ means that Bagging-ICDT significantly outperforms ICDT via the Corrected Paired t-test in the corresponding dataset; • indicates that ICDT performs significantly better than Bagging-ICDT according to the Corrected Paired t-test in the dataset of the row.

Dataset	ICDT	Bagging-ICDT	
anneal	1.7825	**1.7847**	
arrhythmia	1.7861	**1.9316**	○
audiology	2.5156	**2.5936**	
autos	1.4535	**1.5553**	○
balance-scale	**0.6033**	0.6006	
bridges-version1	1.0247	**1.0446**	
bridges-version2	0.8755	**0.9767**	
car	1.2330	**1.2568**	○
cmc	0.2599	**0.2636**	
dermatology	1.6637	**1.6844**	
ecoli	1.6128	**1.6182**	
flags	1.0322	**1.1398**	
hypothyroid	1.3744	1.3744	
iris	0.9911	**0.9982**	
letter	2.5135	**2.6771**	○
lymphography	0.8857	**0.9417**	
mfeat-pixel	1.7194	**2.0066**	○
nursery	1.5350	**1.5398**	○
optdigits	1.7275	**1.9579**	○
page-blocks	1.5332	**1.5418**	○
pendigits	2.0042	**2.0925**	○
postoperative-patient-data	**0.6213**	0.6207	
primary-tumor	1.1476	**1.2278**	
segment	1.8119	**1.8331**	○
soybean	2.7004	**2.7203**	
spectrometer	1.7353	**1.9527**	○
splice	0.9784	**1.0077**	○
sponge	0.9822	**1.0121**	
tae	**0.2218**	0.2216	
vehicle	0.8171	**0.8372**	
vowel	1.7889	**1.8594**	○
waveform	0.6656	**0.7325**	○
wine	0.9658	**0.9817**	
zoo	1.8532	**1.8578**	
Average	1.3652	**1.4248**	

Table 10.12: Summary of the results obtained by ICDT and Bagging-ICDT for the DACC and MIC measures. In the "Wilcoxon test" rows, when one classifier significantly outperforms the other one via the Wilcoxon test, it is expressed by "*". The rows "Paired t-test" indicate the number of datasets where the algorithm in the column performs significantly better than the other one according to the Corrected-Paired t-test.

		ICDT	Bagging-ICDT
DACC:	Average	0.7763	0.8023
	Wilcoxon test		*
	Paired t-test	0	16
MIC:	Average	1.3652	1.4248
	Wilcoxon t-test		*
	Paired t-test	0	14

Table 10.13: Average results obtained by ICDT and Bagging-ICDT for basic metrics. The best scores are marked in bold. Ind size = Indeterminacy size.

Algorithm	Determinacy	Single Accuracy	Set Accuracy	Ind size
ICDT	0.9477	0.8023	**0.8877**	5.2290
Bagging-ICDT	**0.9965**	**0.8037**	0.7792	**2.7013**

10.5 The new cost-sensitive Imprecise Credal Decision Tree

Our proposed cost-sensitive Imprecise Credal Decision Tree combines the idea of weighing instances of the existing Weighted-DT for precise classification, exposed in Section 4.8.1, with the A-NPI-M. We call our proposed method the Weighted Imprecise Credal Decision Tree (Weighted-ICDT).

Let N_{tr} denote the number of instances in the training set and $n_{tr}(c_j)$ the number of training instances that satisfy $C = c_j, \quad \forall j = 1, 2, \ldots, K$. Let us consider the set of A-NPI-M probability intervals on C on the training set:

$$\begin{gathered}\mathcal{I}^{tr}_{ANPI} = \left\{ I^{tr}_{ANPI}(c_j) = \left[\max\left(\frac{n_{tr}(c_j) - 1}{N_{tr}}, 0 \right), \right.\right. \\ \left.\left. \min\left(\frac{n_{tr}(c_j) + 1}{N_{tr}}, 1 \right) \right], \quad j = 1, 2, \ldots, K \right\}.\end{gathered} \tag{10.17}$$

The following credal set corresponds to these probability intervals:

$$\mathcal{P}\left(\mathcal{I}^{tr}_{ANPI}\right) = \left\{ p \in \mathcal{P}(C) \mid p(c_j) \in I^{tr}_{ANPI}(c_j), \quad \forall j = 1, 2, \ldots, K \right\}, \tag{10.18}$$

where $\mathcal{P}(C)$ denotes the set of all probability distributions on C.

Uncertainty measures can be applied to this credal set. As pointed out before, the maximum entropy is a well-established uncertainty measure on credal sets as it satisfies the required properties. Thus, we consider the arrangement of the training instances $(\hat{n}_{tr}(c_1), \hat{n}_{tr}(c_2), \ldots, \hat{n}_{tr}(c_K))$ for which the probability distribution that reaches the maximum entropy on $\mathcal{P}\left(\mathcal{I}^{tr}_{ANPI}\right)$ is attained. Let $(\hat{p}_{tr}(c_1), \hat{p}_{tr}(c_2), \ldots, \hat{p}_{tr}(c_K))$ be the probability distribution that obtains the maximum entropy on $\mathcal{P}\left(\mathcal{I}^{tr}_{ANPI}\right)$, which can be obtained via the algorithm proposed in Section 8.5 for obtaining the probability distribution that reaches the maximum entropy value on an A-NPI-M credal set (Algorithm 14). Then, $\hat{n}_{tr}(c_j) = N_{tr} \times \hat{p}_{tr}(c_j), \quad \forall j = 1, 2, \ldots, K$.

The proposed Weighted-ICDT method considers weights for the instances using the error costs, as Weighted-DT. However, while Weighted-DT uses the relative frequencies in the training set, Weighted-ICDT employs the arrangement that leads to the maximum entropy on $\mathcal{P}\left(\mathcal{I}^{tr}_{ANPI}\right)$.

Let $\mathcal{M}$ be the matrix of errors costs of dimension $K \times K$, where the m_{ij} value indicates the cost of predicting, for an instance, the class value c_i when the real class value is $c_j, \quad \forall i, j \in \{1, 2, \ldots, K\}$. It is always satisfied that $m_{ii} = 0, \quad \forall i = 1, 2, \ldots, K$. Weighted-ICDT computes the weight of a training instance with true class value c_j via the following formula:

$$w_j = \text{Cost}(j) \times \frac{N_{tr}}{\sum_{i=1}^{K} \hat{n}_{tr}(c_i) \times \text{Cost}(i)}, \tag{10.19}$$

where $\mathrm{Cost}(j)$ is the cost of misclassifying an instance whose real class value is c_j, determined by Equation (4.24), $\forall j = 1, 2, \ldots, K$.

Let $\mathcal{D}$ denote the subset of the training set associated with a certain node, $N^{\mathcal{D}}$ the number of instances in $\mathcal{D}$ and $n^{\mathcal{D}}(c_j)$ the number of instances in $\mathcal{D}$ for which $C = c_j, \quad \forall j = 1, 2, \ldots, K$. For the split criterion, Weighted-ICDT considers the A-NPI-M probability intervals on C corresponding to $\mathcal{D}$:

$$\begin{gathered}\mathcal{I}^{\mathcal{D}}_{ANPI} = \left\{ I^{\mathcal{D}}_{ANPI}(c_j) = \left[\max\left(\frac{n^{\mathcal{D}}(c_j) - 1}{N^{\mathcal{D}}}, 0\right),\right.\right. \\ \left.\left.\min\left(\frac{n^{\mathcal{D}}(c_j) + 1}{N^{\mathcal{D}}}, 1\right)\right], \quad j = 1, 2, \ldots, K\right\}.\end{gathered} \tag{10.20}$$

The credal set consistent with these intervals is given by:

$$\mathcal{P}\left(\mathcal{I}^{\mathcal{D}}_{ANPI}\right) = \left\{p \in \mathcal{P}(C) \mid p(c_j) \in I^{\mathcal{D}}_{ANPI}(c_j), \quad \forall j = 1, 2, \ldots, K\right\}. \tag{10.21}$$

Let $\left(\hat{n}^{\mathcal{D}}(c_1), \hat{n}^{\mathcal{D}}(c_2), \ldots, \hat{n}^{\mathcal{D}}(c_K)\right)$ be the arrangement of the class values in the node that gives rise to the maximum entropy on $\mathcal{P}\left(\mathcal{I}^{\mathcal{D}}_{ANPI}\right)$.

Then, Weighted-ICDT estimates the probability of each class value in that node through a weighted proportion of instances, as Weighted-DT. Nevertheless, Weighted-ICDT uses the arrangement that reaches the maximum entropy with the A-NPI-M, unlike Weighted-DT, which employs the relative frequencies in the node. So, the probability of the c_j value estimated by Weighted-ICDT in that node is given by:

$$\hat{p}^{\mathcal{D}}(c_j) = \frac{w_j \times \hat{n}^{\mathcal{D}}(c_j)}{\sum_{i=1}^{K} w_i \times \hat{n}^{\mathcal{D}}(c_i)}, \quad \forall j = 1, 2, \ldots, K. \tag{10.22}$$

In this way, Weighted-ICDT computes the uncertainty about the class variable in that node via the Shannon entropy of the probability distribution $\hat{p}$, determined through Equation (10.22):

$$\hat{S}^{\mathcal{D}}\left(C\right) = -\sum_{j=1}^{K} \hat{p}^{\mathcal{D}}(c_j) \log_2 \hat{p}^{\mathcal{D}}(c_j). \tag{10.23}$$

Let X^i be an attribute whose possible values are $\left\{x^i_1, x^i_2, \ldots, x^i_{t_i}\right\}$. The split criterion of Weighted-ICDT is called the *Weighted Information Gain* (*WIG*). It is based on the entropy defined in Equation (10.23) and is given by:

$$WIG^{\mathcal{D}}(C, X^i) = \hat{S}^{\mathcal{D}}\left(C\right) - \sum_{r_i=1}^{t_i} \hat{P}^{\mathcal{D}}(X^i = x^i_{r_i}) \times \hat{S}^{\mathcal{D}}\left(C \mid X^i = x^i_{r_i}\right), \tag{10.24}$$

where $\hat{S}^{\mathcal{D}}\left(C \mid X^i = x^i_{r_i}\right)$ is the entropy of C on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}$, computed by means of Equation (10.23), and $\hat{P}^{\mathcal{D}}(X^i = x^i_{r_i})$ is the probability that $X^i = x^i_{r_i}$ in $\mathcal{D}$, estimated via proportion of weights:

$$\hat{P}^{\mathcal{D}}(X^i = x^i_{r_i}) = \frac{\sum_{j=1}^{K} n^{\mathcal{D}}(x^i_{r_i}, c_j) \times w_j}{\sum_{j=1}^{K} n^{\mathcal{D}}(c_j) \times w_j}, \quad \forall i = 1, 2, \ldots, t, \tag{10.25}$$

$n^{\mathcal{D}}(x^i_{r_i}, c_j)$ being the number of instances in $\mathcal{D}$ that satisfy $X^i = x^i_{r_i}$ and $C = c_j, \quad \forall r_i = 1, 2, \ldots t_i, \quad i = 1, 2, \ldots, t, \quad j = 1, 2, \ldots, K$.

For classifying an instance at a leaf node, Weighted-ICDT computes, for each value of the class variable, a probability interval based on the A-NPI-M lower and upper probabilities that also takes the weight of the class value into account.

Formally, let $N^{\mathcal{L}}$ be the number of instances in a leaf node $\mathcal{L}$ and $n^{\mathcal{L}}(c_j)$ the number of instances in $\mathcal{L}$ for which $C = c_j, \quad \forall j = 1, 2, \ldots, K$. We know that the A-NPI-M lower and upper probabilities of c_j at $\mathcal{L}$ are given by:

$$\begin{aligned} \underline{P}^{\mathcal{L}}_{ANPI}(c_j) &= \max\left(\frac{n^{\mathcal{L}}(c_j) - 1}{N^{\mathcal{L}}}, 0\right), \\ \overline{P}^{\mathcal{L}}_{ANPI}(c_j) &= \min\left(\frac{n^{\mathcal{L}}(c_j) + 1}{N^{\mathcal{L}}}, 1\right), \quad \forall j = 1, 2, \ldots, K. \end{aligned} \tag{10.26}$$

Weighted-ICDT considers, for the lower (upper) probability, the proportion of weights in an arrangement of the class values for which the A-NPI-M lower (upper) probability is attained. Hence, at that leaf node, we have the following probability interval for each class value:

$$\left[\max\left(\frac{w_j \times \left(n^{\mathcal{L}}(c_j) - 1\right)}{W^{\mathcal{L}}}, 0\right), \min\left(\frac{w_j \times \left(n^{\mathcal{L}}(c_j) + 1\right)}{W^{\mathcal{L}}}, 1\right)\right], \ \forall j = 1, 2, \ldots, K, \tag{10.27}$$

where $W^{\mathcal{L}}$ denotes the sum of all weights at $\mathcal{L}$, that is, $W^{\mathcal{L}} = \sum_{i=1}^{K} w_i \times n^{\mathcal{L}}(c_i)$.

Then, a dominance criterion is applied to these probability intervals to obtain the non-dominated states set. Our proposed Weighted-ICDT algorithm utilizes the stochastic dominance criterion, on these intervals. According to that criterion, a class value c_j dominates another one c_k if, and only if,

$$\begin{aligned}
&\max\left(\frac{w_j \times (n^{\mathcal{L}}(c_j)-1)}{W^{\mathcal{L}}},0\right) \geqslant \min\left(\frac{w_k \times (n^{\mathcal{L}}(c_k)+1)}{W^{\mathcal{L}}},1\right) \Leftrightarrow \\
&\frac{w_j \times (n^{\mathcal{L}}(c_j)-1)}{W^{\mathcal{L}}} \geqslant \frac{w_k \times (n^{\mathcal{L}}(c_k)+1)}{W^{\mathcal{L}}} \Leftrightarrow \\
&w_j \times \left(n^{\mathcal{L}}(c_j)-1\right) \geqslant w_k \times \left(n^{\mathcal{L}}(c_k)+1\right), \quad \forall j,k \in \{1,2,\ldots,K\}.
\end{aligned}$$

Consequently, the non-dominated states set predicted by Weighted-ICDT at $\mathcal{L}$ is determined as follows:

$$\begin{aligned}
&nds^{\mathcal{L}}_{Weighted\_ICDT} = \Big\{c_k,\ 1 \leqslant k \leqslant K \mid w_k \times \left(n^{\mathcal{L}}(c_k)+1\right) > \\
&w_j \times \left(n^{\mathcal{L}}(c_j)-1\right), \quad \forall j = 1,2,\ldots,K\Big\}.
\end{aligned} \tag{10.28}$$

In order to classify an instance with Weighted-ICDT, a path from the root node to a terminal one is made by using the attribute values of such an instance. The predicted non-dominated states set for the instance is the one associated with such a terminal node. The procedure to classify an instance with Weighted-ICDT is summarized in Algorithm 19.

Algorithm 19: Procedure to classify an instance with Weighted-ICDT.

```
Procedure Classify_Weighted_ICDT(Weighted_ICDT $\mathcal{T}$, instance with attribute vector $\mathbf{x}$)
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the attribute vector $\mathbf{x}$.
2. $h^{Weighted\_ICDT}(\mathbf{x}) = nds^{\mathcal{L}}_{Weighted\_ICDT}$, where $nds^{\mathcal{L}}_{Weighted\_ICDT}$ denotes the non-dominated states set predicted by Weighted-ICDT at $\mathcal{L}$, determined via Equation (10.28).
return $h^{Weighted\_ICDT}(\mathbf{x})$
```

10.5.1 Justification of Weighted-ICDT

The most relevant issues of our proposed Weighted-ICDT method can be summarized in the following way:

- Similar to Weighted-DT, Weighted-ICDT computes a weight for each instance depending on the cost of misclassifying the corresponding class value. Both methods estimate the costs in the same way. Nevertheless, whereas Weighted-DT estimates the instance weights based on such costs by using the class frequencies in the training set, Weighted-ICDT utilizes the arrangement that reaches the maximum entropy on the A-NPI-M credal set. Therefore, unlike Weighted-DT, Weighted-ICDT considers that the training set is not totally reliable and employs the well-established uncertainty measure on credal sets.

- For the split criterion, the existing CS-ICDT method considers that all instances have the same importance, regardless of their class values. It is oriented to minimize the number of classification errors but it neglects varying costs of errors. In contrast, for computing the uncertainty of the class variable in a certain node, Weighted-ICDT considers the proportion of instance weights for each class value. In this way, in Weighted-ICDT, the instances whose class value has a higher cost of misclassification have more importance. For example, suppose that we have two class values, c_1 and c_2, where the cost of erroneously classifying an instance with real class value c_1 is ten times the cost of misclassifying an instance whose real class value is c_2. Suppose that, in a certain node, there are four instances whose true class value is c_1 and another four instances with real class value c_2. In this case, for the split criterion, CS-ICDT considers that there is total uncertainty about the class variable, while our proposed Weighted-ICDT algorithm estimates that, in that node, the uncertainty of the class variable is considerably low. So, the uncertainty value estimated by our proposal is intuitively far more reasonable than the one estimated by CS-ICDT because, if that node were terminal, it would be quite logical to predict c_1.

- Indeed, Weighted-DT also considers that the importance of an instance for calculating the uncertainty value depends on the error cost of the associated class value. However, for estimating the probability of each class value, Weighted-ICDT employs the arrangement that obtains the maximum entropy with the A-NPI-M, whereas Weighted-DT uses the arrangement associated with relative frequencies. In consequence, unlike Weighted-DT, Weighted-ICDT considers that the dataset in a certain node is not totally reliable and employs the well-established uncertainty measure on the corresponding A-NPI-M credal set.

- To classify an instance at a leaf node, our proposed Weighted-ICDT method considers, for each class value, a probability interval that depends on the frequency of that class value at that terminal node and the cost of incorrectly classifying an instance that has such a class value. Thus, as in the split criterion, the instances whose class value has a higher cost of erroneous classification have more importance. In contrast, CS-ICDT estimates the lower and upper probabilities for each class value by considering that all instances have the same weight. Afterwards, it computes a risk interval for each class value in which the lower (upper) risk is calculated by considering the lower (upper) probabilities of the remaining class values and the costs of predicting that class value when the real class value is another one. Thereby, the lower and upper probabilities of the corresponding class value do not directly influence the computation of the risk interval, and the cost of misclassifying an instance with that class value is also not taken into account. Concerning the dominance criterion on the probability intervals at leaf nodes, Weighted-ICDT uses the stochastic dominance criterion, the well-established dominance criterion on a given set of probability intervals.

For these reasons, the risk intervals computed by CS-ICDT might be generally less informative than the probability intervals computed by Weighted-ICDT. This issue is illustrated in Example 10.5.1.

- The original CS-ICDT method uses the IDM for the uncertainty measures in the split criterion and for the lower and upper probabilities at leaf nodes. In contrast, our proposed Weighted-ICDT algorithm utilizes the A-NPI-M for estimating the instance weights, in the split criterion, and for the probability intervals at leaf nodes. As commented before, unlike the IDM, the A-NPI-M does not assumes previous knowledge about the data via a parameter, and, consequently, the latter model is more appropriate than the former.

Example 10.5.1 *Suppose that we have a training set of* $N_{tr} = 150$ *instances. Let* C *be the class variable and* $\{c_1, c_2, c_3\}$ *its possible values. Let* $\mathcal{M}$ *denote the matrix of error costs, where* $m_{ii} = 0 \quad \forall i = 1,2,3, \quad m_{i1} = 1$ *for* $i = 2,3, \quad m_{i2} = 2$ *for* $i = 1,3, \quad$ *and* $m_{i3} = 3$ *for* $i = 1,2$.

The costs of misclassifying each class value are given by:

$$\text{Cost}(1) = m_{21} + m_{31} = 2,$$

$$\text{Cost}(2) = m_{12} + m_{32} = 4,$$

$$\text{Cost}(3) = m_{13} + m_{23} = 6.$$

Let us assume the following class frequencies in the training set: $n_{tr}(c_1) = n_{tr}(c_2) = n_{tr}(c_3) = 50$.

In this case, the arrangement that attains the maximum entropy with the A-NPI-M coincides with the one corresponding to relative frequencies. Therefore, the instance weights computed by Weighted-ICDT for the class values are the following ones:

$$w_1 = \mathrm{Cost}(1) \times \frac{N_{tr}}{\sum_{i=1}^{3} \mathrm{Cost}(i) \times n_{tr}(c_i)} = \frac{2 \times 150}{100+200+300} = 0.5,$$

$$w_2 = \mathrm{Cost}(2) \times \frac{N_{tr}}{\sum_{i=1}^{3} \mathrm{Cost}(i) \times n_{tr}(c_i)} = \frac{4 \times 150}{100+200+300} = 1,$$

$$w_3 = \mathrm{Cost}(3) \times \frac{N_{tr}}{\sum_{i=1}^{3} \mathrm{Cost}(i) \times n_{tr}(c_i)} = \frac{6 \times 150}{100+200+300} = 1.5.$$

Suppose that, at a certain leaf node $\mathcal{L}$, $n^{\mathcal{L}}(c_1) = n^{\mathcal{L}}(c_2) = n^{\mathcal{L}}(c_3) = 3$. In such a case, $\underline{P}^{\mathcal{L}}_{ANPI}(c_i) = \frac{2}{9}$ and $\overline{P}^{\mathcal{L}}_{ANPI}(c_i) = \frac{4}{9}$, for $i = 1, 2, 3$. The risk intervals determined by CS-ICDT at $\mathcal{L}$ are given by:

$$\underline{R}_{CS-ICDT}(c_1) = \underline{P}^{\mathcal{L}}_{ANPI}(c_2)m_{12} + \underline{P}^{\mathcal{L}}_{ANPI}(c_3)m_{13} = \frac{10}{9},$$

$$\overline{R}_{CS-ICDT}(c_1) = \overline{P}^{\mathcal{L}}_{ANPI}(c_2)m_{12} + \overline{P}^{\mathcal{L}}_{ANPI}(c_3)m_{13} = \frac{20}{9},$$

$$\underline{R}_{CS-ICDT}(c_2) = \underline{P}^{\mathcal{L}}_{ANPI}(c_1)m_{21} + \underline{P}^{\mathcal{L}}_{ANPI}(c_3)m_{23} = \frac{8}{9},$$

$$\overline{R}_{CS-ICDT}(c_2) = \overline{P}^{\mathcal{L}}_{ANPI}(c_1)m_{21} + \overline{P}^{\mathcal{L}}_{ANPI}(c_3)m_{23} = \frac{16}{9},$$

$$\underline{R}_{CS-ICDT}(c_3) = \underline{P}^{\mathcal{L}}_{ANPI}(c_1)m_{31} + \underline{P}^{\mathcal{L}}_{ANPI}(c_2)m_{32} = \frac{6}{9},$$

$$\overline{R}_{CS-ICDT}(c_3) = \overline{P}^{\mathcal{L}}_{ANPI}(c_1)m_{31} + \overline{P}^{\mathcal{L}}_{ANPI}(c_2)m_{32} = \frac{12}{9}.$$

In consequence, $\overline{R}_{CS-ICDT}(c_i) > \underline{R}_{CS-ICDT}(c_j) \quad \forall i, j \in \{1, 2, 3\}$ and, thus, any of the class values is dominated under the stochastic dominance criterion on these risk intervals.

Regarding Weighted-ICDT, it holds that:

$$w_1 \times \left(n^{\mathcal{L}}(c_1) - 1\right) = 0.5 \times 2 = 1, \quad w_1 \times \left(n^{\mathcal{L}}(c_1) + 1\right) = 0.5 \times 4 = 2,$$

$$w_2 \times \left(n^{\mathcal{L}}(c_2) - 1\right) = 1 \times 2 = 2, \quad w_2 \times \left(n^{\mathcal{L}}(c_2) + 1\right) = 1 \times 4 = 4,$$

$$w_3 \times \left(n^{\mathcal{L}}(c_3) - 1\right) = 1.5 \times 2 = 3, \quad w_3 \times \left(n^{\mathcal{L}}(c_3) + 1\right) = 1.5 \times 4 = 6,$$

Thereby, according to the stochastic dominance criterion utilized in Weighted-ICDT, c_1 *is dominated by both* c_2 *and* c_3*. The non-dominated states set predicted by Weighted-ICDT is* $\{c_2, c_3\}$*.*

Hence, in this situation, the prediction made by Weighted-ICDT is more informative and intuitive than the one made by CS-ICDT.

Table 10.14 summarizes the differences between Weighted-DT, the existing CS-ICDT, and our proposed Weighted-ICDT. It should be noted that, in the proposed Weighted-ICDT method, the weight of an instance for the split criterion depends on the error cost of its class value, unlike CS-ICDT; the criterion used by Weighted-ICDT to classify instances at leaf nodes may be more effective than the one employed by CS-ICDT because the predicted intervals are probably more informative. For these reasons, it is expected that Weighted-ICDT performs better than CS-ICDT. This point is corroborated in Section 10.5.2 with exhaustive experimentation.

Table 10.14: Summary of the differences between Weighted-DT, CS-ICDT, and Weighted-ICDT.

Property	Weighted-DT	CS-ICDT	Weighted-ICDT
Mathematical model	precise probabilities	IDM	A-NPI-M
Error costs in the split criterion	yes	no	yes
Criterion to classify instances	precise prediction	little informative	very informative

10.5.2 Experimental analysis

10.5.2.1 *Experimental setup*

For our experimentation, we take as a reference the experimental study carried out by Abellán and Masegosa in [10], where the ICDT algorithm and its adaptation for cost-sensitive scenarios were proposed.

- **Datasets**: In this experimental analysis, we have employed the same 34 classification datasets used in the other experimental studies carried out in this chapter to test the performance of our proposed Imprecise Classification algorithms. The most important characteristics of these datasets are shown in Table 10.1.

- **Preprocessing**: We have replaced missing values with mean values for continuous attributes and modal values for discrete features. After that, we have discretized continuous attributes by following Fayyad and Irani's discretization method.

- **Algorithms**: Three algorithms have been used in this experimental study: The original CS-ICDT method (CS-ICDT-IDM), a new version of the CS-ICDT algorithm that uses the A-NPI-M instead of the IDM for the split criterion and for the probability intervals at leaf nodes (CS-ICDT-ANPI)[4], and our proposed Weighted-ICDT method. Remark that the computational complexity of these three methods is similar since they are Decision Trees whose split criterion is based on the maximum entropy on a mathematical model based on reachable probability intervals. We do not use more algorithms because, as explained previously, CS-ICDT-IDM and the adaptation of NCC for cost-sensitive scenarios are the only methods for cost-sensitive Imprecise Classification proposed so far, and, since the former algorithm significantly outperforms the latter, considering the adaptation of NCC could introduce noise in the statistical comparisons.

- **Cost matrices**: Let $\sigma : \{1,2,\ldots,K\} \rightarrow \{1,2,\ldots,K\}$ be a permutation that yields a decreasing order of the frequencies of the class values in the training set, i.e $n_{tr}\left(c_{\sigma(i)}\right) \geqslant n_{tr}\left(c_{\sigma(j)}\right) \quad \forall 1 \leqslant i \leqslant j \leqslant K$. The cost matrices used in this experimentation are the following ones:

  - **Cost Matrix** $0/1$: The costs of all erroneous predictions are equal to 1, i.e:

$$\begin{aligned} m_{ij}^{01} &= 1 \quad \forall i,j \in \{1,2,\ldots,K\},\, j \neq i, \\ m_{jj}^{01} &= 0 \quad \forall j = 1,2,\ldots,K. \end{aligned}$$

  - **Cost Matrix (I)**: The cost of an incorrect prediction only depends on the real class value. The class values with lower frequencies have more costs than the ones with higher frequencies. Specifically, the cost of misclassifying an instance whose real class value is the one with the highest frequency is equal to 1, the cost of misclassifying an instance whose true class value is the one with the second-highest frequency is equal to 2, and so on. Formally:

$$\begin{aligned} m_{i\sigma(j)}^{I} &= j \quad \forall i,j \in \{1,2,\ldots,K\},\, \sigma(j) \neq i, \\ m_{jj}^{I} &= 0 \quad \forall j = 1,2,\ldots,K. \end{aligned}$$

[4] It is the adaptation of the ICDT-ANPI method, proposed in Section 10.2, for cost-sensitive classification. Such an adaptation uses the same dominance criterion on risk intervals as CS-ICDT.

- **Cost Matrix (II)**: Only the predicted class value influences the cost of an erroneous prediction. Again, the class values with lower frequencies have more costs than the ones with higher frequencies. The cost of erroneously predicting the class value with the highest frequency is equal to 1, the cost of incorrectly predicting the class value with the second-highest frequency is equal to 2, and so on:

$$\begin{aligned} m^{II}_{\sigma(j)i} &= j \quad \forall i, j \in \{1, 2, \dots, K\},\ \sigma(j) \neq i, \\ m^{II}_{jj} &= 0 \quad \forall j = 1, 2, \dots, K. \end{aligned}$$

- **Cost Matrix (III)**: This cost matrix is equivalent to Cost Matrix (I), but now the class values with lower frequencies have lower costs than the class values with higher frequencies:

$$\begin{aligned} m^{III}_{i\sigma(j)} &= K - j + 1 \quad \forall i, j \in \{1, 2, \dots, K\},\ \sigma(j) \neq i, \\ m^{III}_{jj} &= 0 \quad \forall j = 1, 2, \dots, K. \end{aligned}$$

- **Cost Matrix (IV)**: It is similar to Cost Matrix (II). However, now the class values with lower frequencies have lower costs than the class values with higher frequencies:

$$\begin{aligned} m^{IV}_{\sigma(j)i} &= K - j + 1 \quad \forall i, j \in \{1, 2, \dots, K\},\ \sigma(j) \neq i, \\ m^{IV}_{jj} &= 0 \quad \forall j = 1, 2, \dots, K. \end{aligned}$$

- **Evaluation**: In order to check the performance of the algorithms considered in this experimentation, we mainly use the MIC measure, the well-established evaluation metric for cost-sensitive imprecise classifiers so far. For a deeper analysis of the behavior of the algorithms, we also consider two metrics to evaluate how informative the predictions are: Determinacy and Single Accuracy, which were described in Section 5.3, and two metrics for checking the costs of incorrect classifications of the algorithms: Single Cost and Set Cost. They are, respectively, the adaptations of the Single Accuracy and Set Accuracy metrics, described in Section 5.3, for cost-sensitive classification.

  Formally, let N_{test} be the number of test instances, h the learned Imprecise Classification model, $h(\mathbf{x}^j)$ the non-dominated states set predicted for the j-th test instance, and α_j the maximum cost of predicting a class value belonging to that set, computed through Equation (5.6).

Single Cost is defined as the average misclassification cost among the instances precisely classified:

$$\text{Single\_Cost}(h) = \frac{1}{N_{precise}} \sum_{j=1, |h(x_j)|=1}^{N_{test}} \alpha_j, \qquad (10.29)$$

where $N_{precise} = \left|\{j \in \{1, 2, \ldots, K\} : |h(\mathbf{x_j})| = 1\}\right|$.

Set Cost measures the average error cost between the instances for which more than a class value is predicted:

$$\text{Set\_Cost}(h) = \frac{1}{N_{imprecise}} \sum_{j=1, |h(x_j)|>1}^{N_{test}} \alpha_j, \qquad (10.30)$$

where $N_{imprecise} = \left|\{j \in \{1, 2, \ldots, K\} : |h(\mathbf{x_j})| > 1\}\right|$.

- **Procedure**: For checking the performance of the algorithms considered in this experimental study, for each preprocessed dataset and cost matrix, a cross-validation procedure of 10 folds has been repeated 10 times.

- **Software** and **Parameters**: We have employed the Weka software for our experiments. We have utilized the implementation available in Weka for ICDT, and we have added the necessary structures and methods for CS-ICDT-IDM, CS-ICDT-ANPI, and Weighted-ICDT. We have used the Weka filters for the preprocessing. Also, for cross-validation, we have employed the functionality available in Weka.

  For the IDM parameter in CS-ICDT-IDM, the value $s = 1$, one of the values recommended in [209], has been used, as in the experimental analysis carried out in [10][5]. The rest of the parameters utilized for all algorithms have been the ones given by the default in Weka.

- **Statistical evaluation**: For each cost matrix, we have three algorithms to compare. Hence, following the recommendations of Demšar [75] for statistical comparisons between the results obtained by three or more algorithms on many datasets, the Friedman test has been used with a level of significance of $\alpha = 0.05$ to compare the performance of the algorithms considered here via the MIC measure. If the null hypothesis of this test is rejected, then the algorithms are compared pairwise via the Nemenyi test. Critical diagrams are used to present the results of these tests.

[5] Experiments have been carried out with $s = 2$, but the obtained results are always worse than with $s = 1$. So, they are not reported.

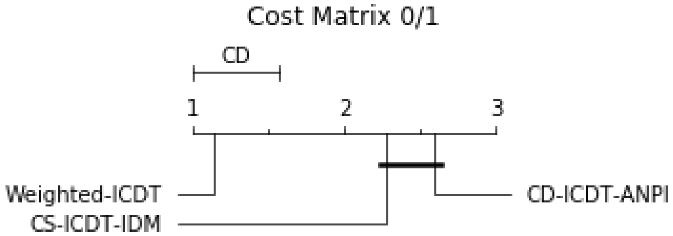

Figure 10.11: Critical diagram for the MIC measure with Cost Matrix 0/1. CD = Critical Distance.

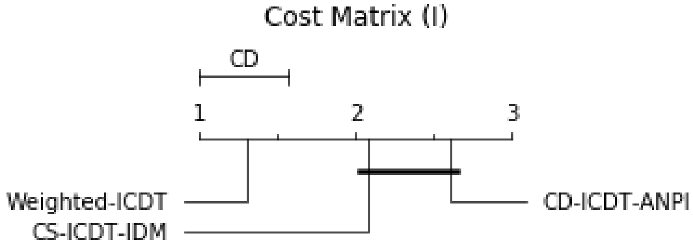

Figure 10.12: Critical diagram for the MIC measure with Cost Matrix (I). CD = Critical Distance.

10.5.2.2 *Results and discussion*

Table 10.15 lets us observe the average Friedman rank obtained by each algorithm for each cost matrix in MIC. The best result for each cost matrix is marked in bold. Figures 10.11, 10.12, 10.13, 10.14, and 10.15 show the critical diagrams for MIC corresponding to Cost Matrices 0/1, (I), (II), (III), and (IV), respectively.

Table 10.15: Average Friedman rank obtained by CS-ICDT-IDM, CS-ICDT-ANPI, and Weighted-ICDT in MIC for each cost matrix.

	Cost Matrix				
Algorithm	0/1	(I)	(II)	(III)	(IV)
CS-ICDT-IDM	2.2794	2.0588	2.2353	2.0882	2.5588
CS-ICDT-ANPI	2.5882	2.6324	2.5441	2.4559	2.6176
Weighted-ICDT	**1.1324**	**1.3088**	**1.2206**	**1.4559**	**1.1765**

We express the following comments about these results:

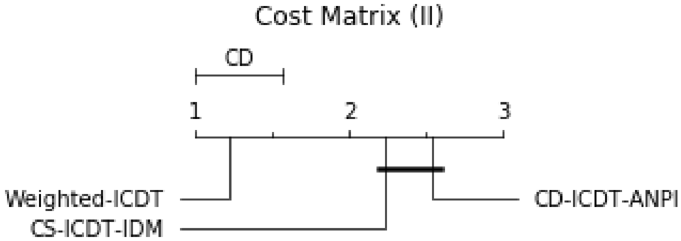

Figure 10.13: Critical diagram for the MIC measure with Cost Matrix (II). CD = Critical Distance.

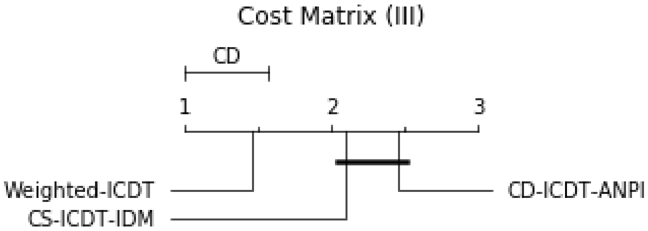

Figure 10.14: Critical diagram for the MIC measure with Cost Matrix (III). CD = Critical Distance.

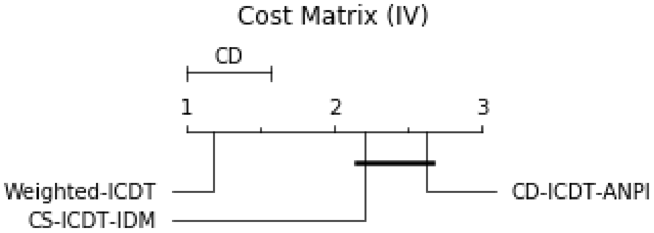

Figure 10.15: Critical diagram for the MIC measure with Cost Matrix (IV). CD = Critical Distance.

- CS-ICDT-IDM obtains a lower average Friedman rank than CS-ICDT-ANPI for all the cost matrices considered in this experimentation. Nevertheless, as CS-ICDT-IDM and CS-ICDT-ANPI are connected via a segment in all critical diagrams, there are no statistically significant differences according to the Nemenyi test between these two algorithms for any of the five cost matrices considered. In consequence, we could state that CS-ICDT-ANPI obtains statistically equivalent results to CS-ICDT-IDM.

- For all cost matrices, the lowest average Friedman rank is achieved by our proposed Weighted-ICDT method. In addition, in the critical diagrams, Weighted-ICDT is not connected with the other two algorithms through segments. Thus, according to the Nemenyi test, Weighted-ICDT performs significantly better than CS-ICDT-ANPI and CS-ICDT-IDM for the five cost matrices. Hence, we can state that Weighted-ICDT achieves, by far, the best results.

Table 10.16 presents, for each cost matrix considered in our experimental analysis, the average results obtained by each algorithm in Determinacy, Indeterminacy Size, Single Cost, and Set Cost. The best result for each measure and cost matrix is marked in bold.

Table 10.16: Average values obtained by CS-ICDT-IDM, CS-ICDT-ANPI, and Weighted-ICDT in the individual evaluation metrics for each cost matrix.

		Cost Matrix				
Measure	Algorithm	0/1	(I)	(II)	(III)	(IV)
Determinacy	CS-ICDT-IDM	0.7094	0.6248	0.6457	0.7159	0.5432
	CS-ICDT-ANPI	0.6835	0.6070	0.6404	0.7025	0.5068
	Weighted-ICDT	**0.9002**	**0.8083**	**0.8756**	**0.8485**	**0.8685**
Single Cost	CS-ICDT-IDM	0.0890	0.1739	0.6457	0.3625	0.1613
	CS-ICDT-ANPI	**0.0806**	**0.1598**	**0.1183**	**0.3584**	**0.1345**
	Weighted-ICDT	0.1521	0.4294	0.4156	0.9636	1.0995
Indeterminacy Size	CS-ICDT-IDM	8.6956	8.7030	**3.6924**	8.4671	**3.4241**
	CS-ICDT-ANPI	8.8938	9.0162	4.4448	8.6728	3.9586
	Weighted-ICDT	**7.9228**	**7.1027**	7.2984	**7.0778**	7.4300
Set Cost	CS-ICDT-IDM	**0.0041**	0.0145	0.3001	**0.0220**	0.8438
	CS-ICDT-ANPI	0.0061	**0.0087**	0.2100	0.0355	0.6742
	Weighted-ICDT	0.0075	0.0993	**0.0636**	0.1074	**0.1528**

These results indicate the following points for each metric:

- **Determinacy**:

  - CS-ICDT-IDM makes more precise predictions than CS-ICDT-ANPI since the average Determinacy value obtained by CS-ICDT-ANPI is lower than the one obtained by CS-ICDT-IDM for all cost matrices.
  - Our proposed Weighted-ICDT algorithm achieves, by far, the highest average Determinacy value for all cost matrices. Hence, this method is, by far, the one that makes more precise predictions among the ones considered in this experimentation.

- **Indeterminacy Size:**
  - For imprecise predictions, CS-ICDT-ANPI predicts more class values than CS-ICDT-IDM due to the average results obtained in Indeterminacy Size. Thus, it can be stated that the predictions made by CS-ICDT-IDM are more informative than the ones made by CS-ICDT-ANPI.
  - The cost matrix influences the average Indeterminacy Size value of Weighted-ICDT. For Cost Matrices 0/1, (I), and (III), Weighted-ICDT obtains the lowest average Indeterminacy Size value. Therefore, for such cost matrices, the imprecise predictions made by Weighted-ICDT are the most informative ones. The opposite happens with Cost Matrices (II) and (IV). Nonetheless, we must remark that, for these cost matrices, the average Determinacy value obtained by Weighted-ICDT is pretty high.

- **Single Cost:**
  - CS-ICDT-ANPI obtains a better result than CS-ICDT-IDM concerning the misclassification cost of precise predictions for all the cost matrices considered here.
  - Weighted-ICDT gets the highest average Single Cost value for the five cost matrices. Thereby, it obtains the highest misclassification costs when predicting a single class value.

- **Set Cost:**
  - For Cost Matrices 0/1, (I), and (III), CS-ICDT-ANPI obtains a higher average Set Cost value than CS-ICDT-IDM, while, for Cost Matrices (II) and (IV), CS-ICDT-ANPI achieves the lowest average Set Cost value. This implies that, for Cost Matrices 0/1, (I), and (III), CS-ICDT-ANPI obtains a higher cost of incorrect imprecise classifications than CS-ICDT-IDM, whereas, for Cost Matrices (II) and (IV), the opposite occurs.

- Weighted-ICDT obtains the highest average Set Cost value for Cost Matrices 0/1, (I), and (III). Hence, for these cost matrices, the cost of incorrect imprecise predictions with Weighted-ICDT is higher than with the other algorithms. The contrary happens with Cost Matrices (II) and (IV).

Summary of the results: The results obtained in this experimental study can be summarized as follows:

- The predictions made by CS-ICDT-ANPI are less informative than the ones made by CS-ICDT-IDM. It is because, as shown in Section 7.3, given a sample of outcomes of a discrete attribute, IDM probability intervals with $s = 1$ are always contained in A-NPI-M probability intervals. Since the predictions made by CS-ICDT-IDM are more precise than the ones made by CS-ICDT-ANPI, the risk of misclassification is higher with the former algorithm and, therefore, the cost of incorrect classifications is generally higher with CS-ICDT-IDM.

- The results obtained in MIC allow deducing that CS-ICDT-IDM and CS-ICDT-ANPI achieve an equivalent trade-off between informative predictions and low misclassification cost and, thus, they perform equivalently. This point is consistent with the experimental study carried out for the proposed ICDT-ANPI (Section 10.2.1), where we showed that the A-NPI-M obtains statistically equivalent results to the IDM with the recommended value of the parameter when both models are utilized in the existing Decision Tree for Imprecise Classification.

- Our proposed Weighted-ICDT method makes much more informative predictions than the other algorithms, even though it leads to a higher cost of incorrect predictions. Weighted-ICDT achieves the best trade-off between informative predictions and low misclassification costs. It is because, as argued in Section 10.5.1, the criterion employed by Weighted-ICDT to classify instances at leaf nodes is probably more effective than the one used by CS-ICDT as the predicted intervals may be more informative and, unlike CS-ICDT, Weighted-ICDT considers the costs of errors for the uncertainty measures in the split criterion.

10.6 Concluding remarks

Classifiers sometimes predict a set of class values since there is not sufficient information to point out a single class value. This is known as Imprecise Classification. The first Imprecise Classification algorithm was the Naïve Credal Classifier (NCC). Afterwards, an Imprecise Classification method based on a single Decision Tree, called the Imprecise Credal Decision Tree (ICDT), was introduced. NCC and ICDT were also adapted for cost-sensitive scenarios.

In this chapter, we have developed important improvement of the algorithms for Imprecise Classification proposed so far. Specifically, we can summarize the contributions of this chapter in the following issues:

- Firstly, we have proposed a new Imprecise Credal Decision Tree that uses the A-NPI-M for the uncertainty measures in the split criterion and the probability intervals at leaf nodes (ICDT-ANPI), unlike the existing ICDT, which employs the IDM. Experimental results have highlighted that the IDM parameter strongly influences the performance of ICDT and that ICDT-ANPI performs equivalently to ICDT with the best choice of the IDM parameter. These results are consistent with the ones obtained by Credal Decision Trees for precise classification.

  Therefore, it can be concluded that the A-NPI-M is more suitable than the IDM to be applied to Decision Trees for Imprecise Classification as the former model does not make prior assumptions about the data and is non-parametric, unlike the latter model.

- We have also developed a new version of the NCC algorithm, called the Extreme Prior Naive Credal Classifier (EP-NCC), that also combines the naïve assumption with the IDM to make imprecise predictions. However, unlike NCC, EP-NCC considers the lower and upper prior probabilities of the class values for the estimation of the lower and upper conditional probabilities. We have shown that our proposed EP-NCC algorithm tends to predict fewer values of the class variable than NCC. In consequence, the predictions made by EP-NCC are more informative. This also implies that, with EP-NCC, there is more risk of making erroneous predictions. Nevertheless, as we have argued, this risk is not far higher than with NCC since EP-NCC narrows the predicted conditional probability intervals by considering the bounds of the prior probabilities of the class values.

  An exhaustive experimental study has been carried out to compare the performance of NCC, our proposed EP-NCC, and the ICDT method.

Such an experimental study has shown that EP-NCC significantly outperforms NCC. Specifically, the predictions made by EP-NCC are far more informative than the ones made by NCC, whereas the difference between both algorithms in making correct predictions is not statistically significant. In addition, ICDT and EP-NCC perform equivalently; while ICDT obtains better results than EP-NCC in the metrics corresponding to Accuracy, the predictions made by EP-NCC are more informative. Nonetheless, ICDT requires a considerably higher computational time than EP-NCC.

For the reasons exposed above, good performance and low computational cost (Table 10.9), it can be concluded that our proposed EP-NCC algorithm is more suitable to be applied to large datasets for Imprecise Classification than the existing algorithms for such a task. Due to the increasing amount of data used in every area, this is a very important point to take into account in favor of EP-NCC.

- Ensemble schemes often improve the performance of individual classifiers in precise classification. None of the Imprecise Classification methods proposed so far makes an ensemble of classifiers. The reason might be that it is not trivial how to combine the predictions made by multiple imprecise classifiers. The first ensemble method for Imprecise Classification has been presented in this chapter. It has been taken into account that the Bagging scheme has been shown to provide pretty good results in precise classification, especially when it has been used with Credal Decision Trees (CDT), which are known to be diverse and unstable classifiers. Hence, the proposed ensemble method consists of a Bagging scheme using the adaptation of CDT for Imprecise Classification (ICDT) as the base classifier. For the combination of the predictions made by multiple imprecise classifiers, we have proposed a new technique that tries that the Bagging imprecise classifier to be as precise as possible. Such a technique consists of predicting as non-dominated only the class values with the lowest possible level of dominance, which implies that it is not very conservative. Reducing the number of non-dominated class values could produce an unnecessary excessive risk.

  Experimental results have revealed that Bagging-ICDT with our proposed combination technique performs much better than the ICDT algorithm. As expected, even though the error rate is a little bit higher for Bagging-ICDT than for ICDT, the former algorithm is much more precise than the latter. In consequence, it can be stated that our developed Bagging method for Imprecise Classification with our proposed technique of combining the predictions made by multiple imprecise classifiers, which tries to maximize *information* assuming more *risk*, is quite appropriate in the sense that it improves the performance of a single ICDT.

- With regard to cost-sensitive classification, in this chapter, we have proposed a new cost-sensitive Imprecise Credal Decision Tree that weights the instances by taking the misclassification cost of the corresponding class value into account. It is based on the idea of an existing Decision Tree for cost-sensitive precise classification. Our new method considers the error costs in the tree-building process, unlike the existing cost-sensitive Imprecise Credal Decision Tree, which only considers the error costs for classifying instances at leaf nodes. Thereby, for the split criterion, an instance has more importance as the misclassification cost of the corresponding class value is higher. In this sense, our proposal presents an advantage over the existing cost-sensitive Imprecise Credal Decision Tree because, in cost-sensitive scenarios, the aim is to minimize the cost of incorrect classifications and not the number of erroneous predictions. Furthermore, our proposed cost-sensitive Imprecise Credal Decision Tree uses the A-NPI-M, whereas the existing one employs the IDM. As explained before, the former model is more suitable than the latter as it does not assume previous knowledge about the data via a parameter. We have also argued that the criterion employed by our proposed cost-sensitive Imprecise Credal Decision Tree to classify instances at leaf nodes is probably more effective than the one used by the existing cost-sensitive Imprecise Credal Decision Tree since the predictions made may be more informative.

  An experimental study has been carried out to check the performance of the existing cost-sensitive Imprecise Credal Decision Tree using the IDM and the A-NPI-M and our proposal. Such an experimental study has revealed that the A-NPI-M obtains statistically equivalent results to the IDM with the recommended value of the parameter when both models are utilized in the existing cost-sensitive Imprecise Credal Decision Tree and, as expected, our proposed cost-sensitive Imprecise Credal Decision Tree performs significantly better than the existing one; even though the cost of erroneous predictions of our proposed method is higher, it is far more informative and achieves a better trade-off between low misclassification cost and informative predictions. Therefore, it can be concluded that our proposed cost-sensitive Imprecise Credal Decision Tree is more suitable than the existing one for practical applications where the misclassification costs are different and the available information is not enough for classifiers to predict a unique class value.

11 IMPRECISE PROBABILITIES IN MULTI-LABEL CLASSIFICATION

11.1 Introduction

In some domains such as *text categorization*, *biology*, or *multimedia*, Multi-Label Classification (MLC) fits better than traditional classification as each instance might belong to multiple labels simultaneously. The MLC problem aims to predict the set of labels corresponding to a certain instance. Many MLC algorithms have been developed so far. A summary of most of them can be found in [97].

On the one hand, the *problem transformation methods* convert the MLC task into multiple traditional classification problems and then combine the solutions of such problems to output a solution for the MLC task. A standard classification algorithm is required to solve the traditional classification problems. Within traditional classification, C4.5 is a well-known method based on Decision Trees. The Credal C4.5 algorithm (CC4.5) [148], exposed in Section 4.6.2, is a version of C4.5 that uses uncertainty measures on credal sets to build the tree. CC4.5 has obtained significantly better results than C4.5 when there is class noise in the data. In this chapter, we analyze the use of CC4.5 in two problem transformation methods: Binary Relevance (BR) and Calibrated Label Ranking (CLR), described in Sections 6.4.1 and 6.4.3, respectively. We argue that the intrinsic label noise in MLC might be higher than the intrinsic class noise in traditional classification. Consequently, CC4.5 is probably more suitable than C4.5 to be employed for solving the binary classification tasks in BR and CC because CC4.5 is less sensitive to class noise than C4.5. Experimental results highlight that CC4.5 performs better than C4.5 when both methods are used to solve the binary classification tasks in BR and CC, the improvement being more notable as there is more noise in the labels.

On the other hand, the *algorithm adaptation methods* directly adapt the existing traditional classification algorithms for MLC. Decision Trees were adapted for MLC by Clare [56]. Such an adaptation, described in Section 6.6, uses precise probabilities for building the tree. Also, multiple versions of the Nearest Neighbors algorithms for MLC have been developed so far. The majority of them use statistical estimators from the neighboring instances based on classical probability theory. We described most of the mentioned lazy MLC algorithms in Section 6.7.

In this chapter, we propose a new adaptation of Decision Trees for MLC that uses imprecise probabilities in the tree-building process and for predicting the posterior probabilities about the relevance of the labels for the instances at leaf nodes. We show that our proposed adaptation might be less sensitive to label noise than the one proposed so far, based on classical probability theory. Via an experimental analysis, we highlight that the proposed adaptation of Decision Trees for MLC performs better than the one developed so far, especially when there is noise in the labels. We also propose new lazy approaches to MLC that use imprecise probability models for the statistical estimators based on the neighboring instances. We show that our proposed lazy approaches to MLC are more suitable than the existing ones based on classical probability theory to handle the class-imbalance problem that frequently arises in MLC, especially when data contain label noise. We carry out an experimental study to corroborate this issue.

Moreover, one of the main challenges of the MLC task is exploiting correlations between labels. Actually, this may be very useful for MLC methods since the number of labels in MLC tends to be very high. The Classifier Chains algorithm (CC), exposed in Section 6.4.2, is considered a simple and effective method to exploit label correlations in MLC. As we know, this method considers a binary classification problem per label in which the previous labels according to an established order are utilized as additional predictive attributes. As pointed out before, the label order strongly influences the performance of CC, and there is no way of determining the optimal label order so far. For these reasons, many label ordering methods for CC have been proposed so far. Most of them estimate correlations between labels via precise probabilities.

A new label ordering method for CC that uses imprecise probabilities to estimate the label correlations is proposed in this chapter. It consists of a greedy procedure that, for each candidate label, takes into account the correlations between that label and the ones already inserted in the chain, as well as the correlations between the candidate label and the labels not inserted yet. We show that our proposal presents some advantages over the label ordering methods based on label correlations proposed so far. An experimental analysis demonstrates that our proposed label ordering procedure performs better than the label ordering methods for CC proposed so far based on label correlations.

To summarize, in this chapter, we analyze the use of imprecise probability models in MLC, showing that they are more suitable than classical probability theory since, as we show, the intrinsic label noise in MC may be higher than the intrinsic class noise in traditional classification and imprecise probability models obtain better results than precise probabilities when data contain noise. We show through several experimental studies that imprecise probabilities lead to better results than classical probability theory in MLC, and this point is enhanced when there is label noise in the data.

The remainder of this chapter is organized as follows: In Section 11.2, we analyze the use of the Credal C4.5 algorithm in the Binary Relevance and Calibrated Label Ranking methods. Section 11.3 describes our proposed adaptation of Decision Trees for Multi-Label Classification. The proposed lazy approaches to Multi-Label Classification that employ imprecise probabilities are detailed in Section 11.4. Section 11.5 presents our proposed label ordering procedure for Classifier Chains based on imprecise probabilities. We conclude this chapter in Section 11.6.

Within this chapter, let $\left\{X^1, X^2, \dots, X^d\right\}$ denote the set of predictive attributes and $\mathcal{Y} = \{y_1, y_2, \dots, y_{n_L}\}$ the label set, where $n_L > 1$. Let $Dom(X^i)$ be the domain of the X^i attribute, $\forall i = 1, 2, \dots, d$. Let $\mathcal{D}_{train} = \{(\mathbf{x_i}, \mathbf{Y_i}), \quad i = 1, 2, \dots, N_{tr}\}$ be the training set, where N_{tr} denotes the number of training instances, $\mathbf{x_i}$ the attribute vector of the i-th training instance, and $\mathbf{Y_i} \subseteq \mathcal{Y}$ its label set, $\quad \forall i = 1, 2, \dots, N_{tr}$.

11.2 Analysis of Credal C4.5 in problem transformation methods

11.2.1 Binary Relevance with Credal C4.5

The Binary Relevance method (BR) [36] considers a binary classification problem per label. In it, the attribute space coincides with the original attribute set and the class variable indicates whether the corresponding label is relevant for a given instance. In order to build the mentioned classifiers, we use the CC4.5 algorithm, described in Section 4.6.2.

When it is required to classify an instance, the set of labels predicted as relevant for such an instance is directly derived from the predictions made by the binary classifiers. The same happens with the predicted posterior probabilities about the relevance of the labels for the instance.

Algorithm 20 summarizes the BR method using the CC4.5 algorithm as the base classifier.

As we know, the BR method is a very simple approach to MLC. Despite this, it has obtained good results in practice, comparable with more sophisticated

Algorithm 20: Binary Relevance with Credal C4.5.

```
Procedure BR-CC4.5 Training set $\mathcal{D}_{train} = \{(\mathbf{x_i}, \mathbf{Y_i}), \quad i = 1, 2, \ldots, N_{tr}\}$
for $j = 1$ to $n_L$ do
    Let $\mathcal{D}_j^{BR}$ be the training set obtained from
     $\mathcal{D}_{train} = \{(\mathbf{x_i}, \mathbf{Y_i}), \quad i = 1, 2, \ldots, N_{tr}\}$ via Equation (6.31)
    Build a binary classifier
     $h_j^{BR\_CC45} : (Dom(X^1), Dom(X^2), \ldots, Dom(X^d)) \to \{0, 1\}$ from $\mathcal{D}_j^{BR}$
     using CC4.5. Such a classifier can also be determined through a
     real-valued function
     $f_j^{BR\_CC45} : (Dom(X^1), Dom(X^2), \ldots, Dom(X^d)) \to \mathbb{R}$.
For classifying a new instance with attribute vector $\mathbf{x}$
$h^{BR\_CC45}(\mathbf{x}) \leftarrow \left\{ y_j, 1 \leqslant j \leqslant n_L \mid h_j^{BR\_CC45}(\mathbf{x}) = 1 \right\}$
for $j = 1$ to $n_L$ do
    $f^{BR\_CC45}(\mathbf{x}, y_j) \leftarrow f_j^{BR\_CC45}(\mathbf{x})$
```

MLC algorithms [144]. Nonetheless, BR has two drawbacks: it ignores label correlations, and the binary classification tasks of BR tend to suffer from a class-imbalance problem.

11.2.2 Calibrated Label Ranking with Credal C4.5

The Calibrated Label Ranking algorithm CLR [94] builds a binary classifier for each pair of labels. For this purpose, those instances for which one of the two labels is relevant and the other one irrelevant are utilized. In this work thesis, the CC4.5 algorithm is used to build these classifiers. This yields a label ranking for a given instance. CLR introduces a virtual label for distinguishing between relevant and irrelevant labels. In this way, CLR considers a binary classification problem per label to predict, for a given instance, the relative relevance of the label versus the virtual one or, in other words, whether such a label is relevant for that instance. We also employ the CC4.5 method to build these classifiers.

When an instance is wanted to be classified, for each label, the number of favorable votes in the classifiers corresponding to the pairwise comparisons is counted. For each label, that number of votes is incremented in one if that label is predicted to be more relevant than the virtual one for the instance to classify. This leads to a label ranking for the instance. For obtaining the set of labels predicted as relevant for that instance, the number of votes of the virtual label is considered. Then, a label is predicted as relevant for the instance, if, and only if, its final number of votes is higher than the number of votes of the virtual label.

The CLR method with the CC4.5 algorithm as the base classifier is summarized in Algorithm 21.

Algorithm 21: Calibrated Label Ranking with Credal C4.5.

```
Procedure CLR-CC4.5 (Training set
$\mathcal{D}_{train} = \{(\mathbf{x_j}, \mathbf{Y_j}), \quad j = 1, 2, \ldots, N_{tr}\}$)
for $j = 1$ to $n_L - 1$ do
    for $k = j + 1$ to $n_L$ do
        Let $\mathcal{D}_{jk}^{CLR}$ be the training set obtained from $\mathcal{D}_{train}$ by means of
          Equation (6.44)
        Build a binary classifier
          $h_{jk}^{CLR\_CC45} : (Dom(X^1), Dom(X^2), \ldots, Dom(X^d)) \rightarrow \{0, 1\}$ from
          $\mathcal{D}_{jk}^{CLR}$ using CC4.5.
for $j = 1$ to $n_L$ do
    Let $\mathcal{D}_{j0}^{CLR}$ be the training set obtained from $\mathcal{D}_{train}$ via Equation
      (6.46).
    Build a binary classifier $h_{j0}^{CLR\_CC45}$ from $\mathcal{D}_{j0}^{CLR}$ employing CC4.5.
For classifying a new instance with attribute vector $\mathbf{x}$
for $j = 1$ to $n_L$ do
    $num\_vot_{\mathbf{x}}^{CC45}(y_j) \leftarrow$
      $\sum_{k=1}^{j-1} \left[\left[h_{kj}^{CLR\_CC45}(\mathbf{x}) = 0\right]\right] + \sum_{k=j+1}^{n_L} \left[\left[h_{jk}^{CLR\_CC45}(\mathbf{x})) = 1\right]\right]$
    $final\_votes_{\mathbf{x}}^{CC45}(y_j) \leftarrow num\_vot_{\mathbf{x}}^{CC45}(y_j) + \left[\left[h_{j0}^{CLR\_CC45}(\mathbf{x}) = 1\right]\right]$
    $f^{CLR\_CC45}(\mathbf{x}, y_j) = \frac{final\_votes_{\mathbf{x}}^{CC45}(y_j)}{n_L}$
$votes\_virtual^{CC45}(\mathbf{x}) \leftarrow \sum_{j=1}^{n_L} \left[\left[h_{j0}^{CLR\_CC45}(\mathbf{x}) = 0\right]\right]$.
$h^{CLR\_CC45}(\mathbf{x}) \leftarrow$
  $\{y_j \mid final\_votes_{\mathbf{x}}^{CC45}(y_j) > votes\_virtual^{CC45}(\mathbf{x}), \quad 1 \leqslant j \leqslant n_L\}$
```

As pointed out previously, CLR mitigates the class-imbalance problem that usually appears in MLC and allows exploiting correlations between pairs of labels.

11.2.3 Justification of CC4.5 as the base classifier

As we know, C4.5 is a well-known traditional classification algorithm based on Decision Trees. In this subsection, we study the use of CC4.5 versus C4.5 to tackle the binary classification problems in BR and CLR. We may note the following issues concerning the use of CC4.5 versus C4.5 as the base classifier of BR and CLR:

- **Intrinsic label noise in MLC**: It is known that, in classification datasets, there are often errors although they have not been intentionally added. This is known as *intrinsic noise* and can be due to errors in data extraction or because the value of a variable is not exactly known. In this subsection, we argue in detail that the intrinsic label noise in MLC may be higher than the intrinsic class noise in traditional classification.

  Let p^{noise} be the probability that an instance has an error in the class variable in traditional classification. Suppose that, in MLC, the probability that an instance has an error in a label is also p^{noise} (we are assuming the same probability of error for all labels). In this case, the probability that an instance has the correct class value in traditional classification is

  $$1 - p^{noise}.$$

  In MLC, assuming that there is independence between the errors in the labels, the probability that an instance has no error in any label is equal to

  $$\left(1 - p^{noise}\right)^{n_L}.$$

  Suppose that, in the previous situation, $p^{noise} = 0.05$ and $n_L = 10$. Then, the probability that an instance has the correct class value in traditional classification is equal to

  $$1 - 0.05 = 0.95.$$

  The probability that an instance has the correct value in all labels is

  $$(0.95)^{10} = 0.6$$

  Consequently, in this case, the probability that an instance has an error in some label is 0.4. Thereby, even though the probability that an instance has an error in a specific label is very low, the probability of error in at least one label can be quite notable. In consequence, we can easily deduce that the intrinsic label noise in MLC is probably higher than in traditional classification. For this reason, in MLC, it seems to be appropriate to use classifiers robust to label noise.

- **Sensitivity to label noise**: In [148, 150], it was demonstrated that the split criterion of CC4.5 is more robust to class noise than the split criterion of C4.5. Moreover, in those works, it was empirically shown that CC4.5 and C4.5 have equivalent performance without class noise in the data and that CC4.5 obtains significantly better results than C4.5 when classifying class noisy data. Therefore, it was concluded that CC4.5 is less sensitive to class noise than C4.5.

  When there is an error in a certain label for a given training instance, the corresponding binary classifier of BR has a training instance with an incorrect value of the class variable. Likewise, when a training instance has an error in a label, the binary classifiers associated with the pairwise comparisons in CLR have an error in the class value of that instance. In these cases, BR and CLR are less affected by the noise by using CC4.5 rather than C4.5. Hence, BR and CLR are more robust to label noise with CC4.5 than with C4.5.

To sum up, BR obtains good results in practice despite being really simple; CLR exploits pairwise label correlations and mitigates the class-imbalance problem that often arises in MLC; C4.5 is a well-known traditional classification method; BR and CLR are more robust to label noise with CC4.5 than with C4.5; the intrinsic label noise in MLC might be higher than the intrinsic class noise in traditional classification. For these reasons, it is worth empirically analyzing BR and CLR using CC4.5 as the base classifier, checking whether it supposes an improvement over C4.5.

11.2.4 Experiments

In this experimental study, we aim to compare the performance of C4.5 and CC4.5 when both algorithms are employed to tackle the binary classification tasks of BR and CLR.

11.2.4.1 *Experimental settings*

- **Datasets**: Thirteen datasets have been employed in our experimental analysis. They can be downloaded from the official website of Mulan [202][1], a Java library for MLC. Most of these datasets have been used in other experimental studies for MLC methods [49, 144]. Table 11.1 shows the main characteristics of each dataset: number of instances, number

[1] http://mulan.sourceforge.net/datasets-mlc.html

of continuous and discrete attributes, number of labels, label cardinality, label density, and MLC domain.

Table 11.1: Datasets used in our experimentation with MLC methods. N is the number of instances, N_CA and N_DA are, respectively, the number of continuous and discrete attributes, N_L is the number of labels, L_C the label cardinality, and L_D the label density.

Dataset	N	N_DA	N_CA	N_L	L_C	L_D	Domain
bibtex	7395	1836	0	159	2.4	0.015	Text
birds	645	2	258	19	1.014	0.053	Multimedia
cal500	502	0	68	174	26.044	0.15	Multimedia
corel5k	5000	499	0	374	3.52	0.009	Multimedia
delicious	16105	500	0	983	19.02	0.019	Text
emotions	593	0	72	6	1.87	0.311	Multimedia
enron	1702	1001	0	53	3.38	0.064	Text
flags	194	9	10	7	3.392	0.485	Multimedia
genbase	662	1186	0	27	1.252	0.046	Biology
mediamill	43907	0	120	101	4.38	0.043	Multimedia
medical	978	1449	0	45	1.24	0.028	Text
scene	2407	0	294	6	1.07	0.179	Multimedia
yeast	2417	0	103	14	4.24	0.303	Biology

We must remark that the 'delicious' dataset requires a very high computational cost due to its very large numbers of instances and labels. For this reason, this dataset is only used for BR.

It can be observed that the datasets used in this experimental analysis are diverse with regard to the number of instances, number of continuous and discrete features, number of labels, label cardinality, and label density. Hence, we can state that the set of datasets used in our experimentation is representative.

- **Algorithms**: Two MLC algorithms have been employed in this experimental study: BR and CLR. For both of them, two base classifiers have been considered: C4.5 and CC4.5.

- **Evaluation metrics**: Consistently with the extensive experimental study with MLC algorithms carried out in [144], sixteen evaluation metrics have been used in this experimentation: Six are based on instance-classification: Hamming Loss, Subset Accuracy, Accuracy, Precision, Recall, and F1; six label-based classification evaluation metrics have been employed: Micro Precision, Macro Precision, Micro Recall, Macro Recall, Micro F1, and Macro F1; the other four evaluation measures are based on ranking: Coverage, One Error, Ranking Loss, and Average Precision. All these metrics were exposed in Section 6.3.

- **Procedure**: Three noise levels have been considered in our experiments: 0%, 5%, and 10%. For each MLC algorithm, base classifier, dataset, and noise level, the following cross-validation procedure has been carried out: the dataset has been divided into five partitions, and, for each one of them, an iteration has been done. In it, the associated partition has been employed for testing, and the rest of the data for training. For each label, the x% of the training instances (x being the noise level) have been chosen, and the value of their label has been changed (if the label is irrelevant it has been changed to relevant and vice-versa)[2]. The MLC model is learned with the noisy training set, and the evaluation measures are extracted via the test set. The same partitions have been employed for all combinations of MLC algorithm/base classifier in all datasets.

- **Software** and **parameters**: The implementations available in Mulan for BR and CLR have been used for this experimentation. The implementations given in Weka for both C4.5 and CC4.5 have been employed. For the IDM parameter in CC4.5, the value $s = 1$ has been utilized because it is one of the values recommended by Walley [209], has been used in the experimental study carried out by the developers of CC4.5 in [149], and requires a low computational cost. The rest of the parameters used for all algorithms have been the ones given by default in the corresponding software.

  Part of the functionality available in Mulan has been employed to create the partitions of cross-validation. The Weka filters have been utilized for generating the label noise.

- **Statistical evaluation**: For each MLC algorithm considered here and evaluation metric, we have two base classifiers to compare: C4.5 and CC4.5. In this way, following the indications given in [49, 75] for statistical comparisons between two methods, the Wilcoxon test has been employed with a level of significance of $\alpha = 0.05$ for checking which base classifier achieves better performance and whether the differences are statistically significant.

11.2.4.2 *Results and discussion*

Tables 11.2 and 11.3 show a summary of the results obtained by each base classifier for each evaluation metric and noise level in BR and CLR, respec-

[2] We have not considered noise levels higher than 10% because the noise is introduced in each label and, thus, a higher noise level would imply a very considerable amount of noise in the data.

tively. Specifically, for each metric and noise level, they illustrate which base classifier performs better according to the Wilcoxon test and whether the differences are statistically significant. Also, these tables let us see, for each metric and noise level, in how many datasets a base classifier achieves a better result than the other one (number of wins).

Binary Relevance: The following points should be noted about the results obtained by the BR algorithm for each type of evaluation metric:

- **Instance-based classification metrics**:
  - When there is no label noise in the data, according to the Wilcoxon test, CC4.5 performs significantly better than C4.5 as the base classifier of BR in Hamming Loss and Subset Accuracy. It means that BR predicts the entire sets of relevant labels for the instances with CC4.5 more frequently than with C4.5 and that BR incorrectly classifies fewer pairs of instance-label with CC4.5.
  - Concerning Recall, in ten datasets, C4.5 achieves a better result than CC4.5 as the base classifier of BR, whereas, in only three datasets, the opposite happens. However, in this case, both base classifiers perform equivalently according to the Wilcoxon test.
  - CC4.5 obtains a higher number of wins than C4.5 as the base classifier of BR in Accuracy, Precision, and F1. Consequently, BR generally predicts better the sets of relevant labels for the instances with CC4.5 than with C4.5; BR predicts less irrelevant labels as relevant using CC4.5 than employing C4.5, and the results of BR in the harmonic means between Precision and Recall are more favorable to CC4.5. Nevertheless, there no are statistically significant differences via the Wilcoxon test for none of these three metrics.
  - When there is noise in the labels, CC4.5 significantly outperforms C4.5 via the Wilcoxon test as the base classifier of BR in Hamming Loss, Subset Accuracy, Accuracy, Precision, and F1. Furthermore, in these metrics, the number of wins of CC4.5 is notably higher than the number of wins of C4.5.
  - With noise in the labels, in Recall, there are more datasets in which BR performs better using C4.5 than datasets in which BR gets a better result with CC4.5. Nonetheless, there are no statistically significant differences in Recall via the Wilcoxon test for any of the label noise levels considered.

Table 11.2: Summary of the results obtained by BR with C4.5 and CC4.5 for each evaluation metric and noise level. (•) means that the base classifier of the column performs significantly better than the other one according to the Wilcoxon test. (-) indicates that the algorithm of the column improves the other one in BR, but the differences are not statistically significant.

Noise level	Metric	C4.5	CC4.5	Wins C4.5	Wins CC4.5
0% Noise	Hamming Loss		(•)	3	9
	Subset Accuracy		(•)	3	8
	Accuracy		(-)	5	8
	Precision		(-)	5	8
	Recall	(-)		10	3
	F1		(-)	5	8
	Micro Precision		(-)	5	8
	Macro Precision		(-)	4	9
	Micro Recall	(-)		10	3
	Macro Recall	(-)		8	4
	Micro F1		(-)	4	9
	Macro F1	(-)		8	5
	Coverage		(•)	1	12
	Ranking Loss		(•)	0	12
	Average Precision		(•)	2	11
	One-Error		(•)	2	9
5% Noise	Hamming Loss		(•)	3	9
	Subset Accuracy		(•)	1	10
	Accuracy		(•)	4	8
	Precision		(•)	3	9
	Recall	(-)		9	3
	F1		(•)	4	8
	Micro Precision		(•)	3	9
	Macro Precision		(-)	4	8
	Micro Recall	(•)		10	2
	Macro Recall	(•)		10	2
	Micro F1		(-)	5	7
	Macro F1	(-)		6	6
	Coverage		(•)	3	9
	Ranking Loss		(•)	2	10
	Average Precision		(•)	2	10
	One-Error		(•)	1	11
10% Noise	Hamming Loss		(•)	2	10
	Subset Accuracy		(•)	1	10
	Accuracy		(•)	2	10
	Precision		(•)	2	10
	Recall	(-)		8	4
	F1		(•)	2	10
	Micro Precision		(•)	2	10
	Macro Precision		(-)	4	8
	Micro Recall	(•)	(-)	9	3
	Macro Recall	(•)		11	1
	Micro F1		(•)	4	8
	Macro F1	(-)		8	4
	Coverage		(•)	2	10
	Ranking Loss		(•)	1	11
	Average Precision		(•)	1	11
	One-Error		(•)	0	12

Table 11.3: Summary of the results obtained by CLR with C4.5 and CC4.5 for each evaluation metric and noise level. (•) means that the base classifier of the column performs significantly better than the other one according to the Wilcoxon test. (-) indicates that the algorithm of the column improves the other one in CLR, but the differences are not statistically significant.

Noise level	Metric	C4.5	CC4.5	Wins C4.5	Wins CC4.5
0% Noise	Hamming Loss		(-)	3	7
	Subset Accuracy		(-)	4	6
	Accuracy		(-)	5	7
	Precision	(-)		8	4
	Recall	(-)		7	5
	F1		(-)	5	7
	Micro Precision	(-)		7	5
	Macro Precision		(-)	6	6
	Micro Recall	(-)		7	5
	Macro Recall	(-)		6	5
	Micro F1		(-)	4	8
	Macro F1		(-)	4	8
	Coverage	(-)		6	5
	Ranking Loss	(-)		7	4
	Average Precision	(-)		6	5
	One-Error		(-)	5	6
5% Noise	Hamming Loss		(-)	3	8
	Subset Accuracy		(-)	3	7
	Accuracy		(•)	3	8
	Precision		(-)	6	5
	Recall		(-)	5	6
	F1		(-)	3	8
	Micro Precision		(-)	5	6
	Macro Precision	(-)		6	5
	Micro Recall	(-)		6	5
	Macro Recall	(-)		8	3
	Micro F1		(-)	4	7
	Macro F1	(-)		5	6
	Coverage		(-)	4	8
	Ranking Loss		(-)	4	8
	Average Precision		(-)	3	9
	One-Error		(-)	3	8
10% Noise	Hamming Loss		(•)	2	10
	Subset Accuracy		(•)	0	11
	Accuracy		(•)	0	11
	Precision		(•)	2	10
	Recall		(-)	4	8
	F1		(•)	1	11
	Micro Precision		(-)	4	8
	Macro Precision		(-)	6	6
	Micro Recall		(-)	5	7
	Macro Recall	(-)		8	4
	Micro F1		(•)	2	10
	Macro F1	(-)		7	5
	Coverage		(-)	4	8
	Ranking Loss		(•)	3	9
	Average Precision		(•)	2	9
	One-Error		(•)	2	9

- **Label-based classification metrics**:
  - When there is no noise in the labels, the results obtained by BR with C4.5 and CC4.5 are statistically equivalent according to the Wilcoxon test in all label-based classification evaluation measures.
  - Despite the previous point, without noise in the labels, BR obtains more wins with CC4.5 than with C4.5 in the metrics corresponding to Precision, while, in the metrics associated with Recall, the opposite happens. Therefore, BR predicts fewer irrelevant labels as relevant with CC4.5 than with C4.5 but predicts more relevant labels as relevant with the latter base classifier than with the former.
  - Without noise in the labels, the results obtained by C4.5 and CC4.5 as the base classifiers of BR in Micro and Macro F1 are statistically equivalent according to the Wilcoxon test. The number of wins of CC4.5 is higher than the number of wins of C4.5 in Micro F1, and the contrary happens in Macro F1.
  - Something similar occurs with noise in the labels. However, BR performs significantly better with C4.5 than with CC4.5 in the metrics corresponding to Recall. In consequence, the fact that BR predicts more relevant labels as relevant with C4.5 than with CC4.5 is now enhanced. In contrast, CC4.5 performs significantly better than C4.5 as the base classifier of CLR in Micro Precision. Indeed, with noise in the labels, C4.5 and CC4.5 have equivalent performance in BR according to the Wilcoxon test in Macro Precision. Even so, it is remarkable that, in this case, CC4.5 gets a notably higher number of wins than C4.5. So, with noise in the labels, BR predicts much fewer irrelevant labels as relevant with CC4.5 than with C4.5.
  - With a 10% of noise in the labels, CC4.5 significantly outperforms C4.5 via the Wilcoxon test as the base classifier of BR in Micro Precision. In contrast, with this noise level, there are no statistically significant differences between C4.5 and CC4.5 in Macro F1.
- **Ranking-based measures**: For all noise levels, CC4.5 performs significantly better than C4.5 as the base classifier of BR according to the Wilcoxon test in all ranking-based metrics. In addition, the number of wins of CC4.5 is far higher than the number of wins of C4.5 in all ranking-based evaluation measures for all noise levels. Therefore, BR predicts the posterior probabilities about the relevance of the labels for the instances much better with CC4.5 than with C4.5.

Calibrated Label Ranking:

- **Instance-based classification measures**:
  - Without noise in the labels, there are no statistically significant differences via the Wilcoxon test between the results obtained by C4.5 and CC4.5 as the base classifiers of CLR in any of the instance-based classification metrics considered.
  - However, even without noise in the labels, in Hamming Loss, CLR obtains a better result with CC4.5 in seven datasets and with C4.5 in three datasets. Thereby, without label noise, for a given instance, there is generally less difference between the set of labels associated with an instance and the set of labels predicted as relevant for such an instance with CC4.5 as the base classifier of CLR than with C4.5.
  - When there is no noise in the labels, in Precision, the number of wins of CC4.5 is four over eight of C4.5. Hence, without noise in the labels, with CC4.5 as the base classifier of CLR, more irrelevant labels are predicted as relevant than with C4.5.
  - With a 5% of noise in the labels, CLR performs better with CC4.5 than with C4.5 in all instance-based classification evaluation measures. In Hamming Loss, Subset Accuracy, Accuracy, and F1, the number of datasets in which CC4.5 performs better than C4.5 in CLR is notably larger than the number of datasets in which C4.5 gets a better result. Furthermore, CC4.5 significantly outperforms C4.5 via the Wilcoxon test as the base classifier of CLR in Accuracy. Therefore, with a 5% of label noise, CLR predicts the set of labels associated with the instances much better using CC4.5 as the base classifier rather than C4.5.
  - The differences are more notable when there is a 10% of noise in the labels. In fact, with this noise level, the results obtained by CLR with CC4.5 are significantly better than the ones achieved with C4.5 in all instance-based classification metrics according to the Wilcoxon test, except for Recall. Indeed, the number of wins of CC4.5 is far higher than the number of wins of C4.5 in all these measures, even in Recall, where CLR obtains a better result with CC4.5 in eight datasets and with C4.5 in three datasets. Thus, with a 10% of label noise in the data, CLR predicts the sets of relevant labels for the instances with CC4.5 as the base classifier much better than with C4.5.
- **Label-based classification metrics**:

- As in instance-based classification metrics, when there is no noise in the labels, the results obtained by C4.5 and CC4.5 as the base classifiers of CLR are statistically equivalent according to the Wilcoxon test in all label-based classification measures.
- Nevertheless, in the metrics corresponding to the harmonic means between Precision and Recall (Micro and Macro F1), the performance of CLR with CC4.5 is better than with C4.5, even though we must remark that the differences are not statistically significant via the Wilcoxon test for any of these measures.
- With a 5% of label noise, according to the Wilcoxon test, the differences between the results obtained by CLR with C4.5 and CC4.5 are also not statistically significant for any of the label-based classification metrics considered here.
- However, it is remarkable that, with a 5% of label noise, the results achieved by CLR using C4.5 as the base classifier in Micro and Macro Recall are better than with CC4.5 (eight wins and three losses in both metrics). Consequently, when there is a 5% of label noise, with C4.5, CLR predicts more relevant labels as relevant than with CC4.5, although the differences are not statistically significant.
- With a 10% of label noise, the results are similar to the ones obtained with a 5% of label noise. Nonetheless, according to the Wilcoxon test, the performance of CLR with CC4.5 as the base classifier in the harmonic mean between Micro Precision and Micro Recall, i.e. Micro F1, is significantly better than with C4.5. In consequence, when there is a 10% of label noise, in harmonic means between Precision and Recall averaged over all instance/label pairs, CLR obtains much better performance with CC4.5 than with C4.5.

- **Ranking-based measures**:
  - When there is no noise in the labels, the results obtained by C4.5 and CC4.5 in CLR are statistically equivalent according to the Wilcoxon test in all the ranking-based metrics considered in this experimental analysis, similar to instance-based evaluation measures. We may observe that, in all ranking-based metrics, the numbers of wins of both base classifiers of CLR are similar, except for Ranking Loss. In this metric, CLR performs better with CC4.5 in only three datasets and with C4.5 in seven datasets. Thereby, without noise in the labels, CLR reversely orders fewer pairs of relevant-irrelevant labels

with C4.5 than with CC4.5, even though, in this case, there are no statistically significant differences.

- With a 5% of noise in the labels, according to the Wilcoxon test, there are also no statistically significant differences between the results obtained by C4.5 and CC4.5 as the base classifiers of CLR in any ranking-based evaluation metric. However, in all ranking-based measures, the number of datasets in which CC4.5 obtains a better result than C4.5 as the base classifier of CLR is much higher than the number of datasets where the contrary happens. Hence, with a 5% of noise in the labels, CLR predicts the posterior probabilities of the relevance of the labels for the instances far better with CC4.5 than with C4.5.
- The difference between the performance of CLR with CC4.5 and C4.5 in ranking-based metrics is even more notable with a 10% of label noise. Indeed, CC4.5 significantly outperforms C4.5 via the Wilcoxon test as the base classifier of CLR in all the ranking-based metrics considered in this experimentation, except for Coverage, which measures the number of steps, on average, that are required to go down the label ranking for covering all labels associated with an instance. Even so, in this evaluation metric, CC4.5 gets eight wins and three losses. In this way, when there is a 10% of label noise, the posterior probabilities of the relevance of the labels for the instances predicted by CLR employing CC4.5 as the base classifier are far more suitable than using C4.5.

Summary of the results: The following issues summarize the results obtained in this experimental analysis:

- When there is no noise in the labels, both BR and CLR perform better using CC4.5 as the base classifier than with C4.5. Actually, in both MLC algorithms, the number of wins of CC4.5 is higher than the number of wins of C4.5 in most of the evaluation metrics. Furthermore, in BR, in some evaluation metrics, CC4.5 significantly outperforms C4.5 via the Wilcoxon test. This happens because the intrinsic label noise in MLC is probably higher than in traditional classification and CC4.5 is more robust to class noise than C4.5.

- As the level of label noise is higher, there are more datasets where CC4.5 achieves a better result than C4.5 and more evaluation measures in which, according to the Wilcoxon test, CLR obtains significantly better performance using CC4.5 than with C4.5. So, the improvement of CC4.5 over C4.5 as the base classifier of BR and CLR is more notable as there is more noise in the labels. We already know the reason: C4.5 is more sensitive to class noise than CC4.5 and, thus, BR and CLR are more robust to label noise with CC4.5 than with C4.5.

- We appreciate that, in BR, C4.5 performs better than CC4.5 in the metrics corresponding to Recall whereas CC4.5 outperforms C4.5 in the metrics associated with Precision. This means that BR predicts more relevant labels as relevant with C4.5 as the base classifier than with CC4.5, but BR predicts fewer irrelevant labels as relevant with the latter base classifier than with the former. It occurs since, as commented before, the binary classification problems in BR tend to suffer from a class-imbalance problem, and CC4.5 stops branching the tree before C4.5. Consequently, in many cases, with CC4.5, some parts of the tree in which a label is correctly predicted as relevant with C4.5 are not reached. However, in such parts, the label noise may have a negative influence. The results obtained in F1 metrics let us deduce that the trade-off between correctly predicting relevant labels and not predicting irrelevant labels as relevant is better for CC4.5, especially when there is noise in the labels.

- The improvement is especially notable in instance-based and ranking-based evaluation metrics. It is because the proportion of instances with an error in at least one label is probably far higher than the proportion of instances that have an error in a specific label.

11.3 Multi-Label Credal Decision Tree

The adaptation of Decision Trees for MLC proposed here, called the Multi-Label Credal Decision Tree (ML-CDT), principally differ from the adaptation proposed so far, described in Section 6.6, in the split criterion and in the criterion used to predict the posterior probabilities about the relevance of the labels for the instances at leaf nodes.

Let $\mathcal{D}$ be the subset of the training set corresponding to a certain node and $N^{\mathcal{D}}$ the total number of instances in $\mathcal{D}$. Let $n^{\mathcal{D}}(y_j)$ ($n^{\mathcal{D}}(\overline{y_j})$) denote the number of instances in $\mathcal{D}$ for which y_j is relevant (irrelevant), $\quad \forall j = 1, 2, \ldots, n_L$.

For each label y_j, we have the following A-NPI-M probability interval on y_j associated with $\mathcal{D}$:

$$I^{\mathcal{D}}_{ANPI}(y_j) = \left[\max\left(\frac{n^{\mathcal{D}}(y_j)-1}{N^{\mathcal{D}}}, 0\right), \min\left(\frac{n^{\mathcal{D}}(y_j)+1}{N^{\mathcal{D}}}, 1\right)\right], \quad \forall j = 1, 2, \ldots, n_L. \tag{11.1}$$

This probability interval leads to the following credal set on y_j on $\mathcal{D}$:

$$\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right) = \left\{p \in \mathcal{P}(y_j) \mid p(y_j) \in I^{\mathcal{D}}_{ANPI}(y_j)\right\}, \tag{11.2}$$

where $\mathcal{P}(y_j)$ denotes the set of all probability distributions on y_j and $p(y_j)$ is the probability that y_j is relevant according to the probability distribution p.

As we know, the maximum entropy is a well-established uncertainty measure on credal sets as it satisfies the essential mathematical properties. Hence, ML-CDT considers the maximum entropy on $\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)$:

$$S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) = \max_{p \in \mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)} S(p), \tag{11.3}$$

$S(p)$ being the Shannon entropy of the probability distribution p.

Computing $S^*\left(\mathcal{P}\left(I^{\mathcal{D}}(y_j)\right)\right)$ is direct. Indeed, applying Algorithm 14, we obtain that the probability distribution that gives rise to the maximum entropy on $\mathcal{P}\left(I^{\mathcal{D}}(y_j)\right)$, $\hat{p}^{\mathcal{D}}_j$, is given by:

$$\hat{p}^{\mathcal{D}}_j(y_j) = \begin{cases} \frac{1}{2} & \text{if } \left|n^{\mathcal{D}}(y_j) - n^{\mathcal{D}}(\overline{y_j})\right| \leqslant 2 \\ \frac{n^{\mathcal{D}}(y_j)-1}{N^{\mathcal{D}}} & \text{if } n^{\mathcal{D}}(y_j) > n^{\mathcal{D}}(\overline{y_j}) + 2 \\ \frac{n^{\mathcal{D}}(y_j)+1}{N^{\mathcal{D}}} & \text{if } n^{\mathcal{D}}(\overline{y_j}) > n^{\mathcal{D}}(y_j) + 2 \end{cases} \tag{11.4}$$

The basis of the split criterion of ML-CDT is the maximum entropy on the entire label set $\mathcal{Y}$, which is determined by the sums of the maximum entropies on the A-NPI-M credal sets on the labels:

$$S^*(\mathcal{Y}) = \sum_{j=1}^{n_L} S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right). \tag{11.5}$$

For the split criterion, ML-CDT considers the gain of information of the label set assuming that the entropy of the label set is computed via the sum of the maximum entropies on the A-NPI-M credal sets on the labels. Formally, for an attribute X^i whose possible values are $\{x^i_1, x^i_2, \ldots, x^i_{t_i}\}$, the split criterion of ML-CDT is defined in the following way:

$$IIG^{\mathcal{D}}(\mathcal{Y}) = S^*(\mathcal{Y}) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) S^*\left(\mathcal{Y} \mid X^i = x^i_{r_i}\right), \tag{11.6}$$

where $P^{\mathcal{D}}(X^i = x^i_{r_i})$ is the probability that $X^i = x^i_{r_i}$ in $\mathcal{D}$, estimated through relative frequencies, and $S^*\left(\mathcal{Y} \mid X^i = x^i_{r_i}\right)$ is the maximum entropy on $\mathcal{Y}$ on the subset of $\mathcal{D}$ composed of those instances for which $X^i = x^i_{r_i}$, computed via Equation (11.5), $\quad \forall r_i = 1, 2, \ldots, t_i, \quad i = 1, 2, \ldots, d$.

For classifying an instance at a leaf node, ML-CDT predicts the posterior probability about the relevance of a label at that leaf node through the probability distribution that reaches the maximum entropy on the corresponding A-NPI-M credal set on such a label. A label is predicted as relevant at that leaf node if, and only if, its predicted posterior probability is greater or equal than 0.5. Formally, let $\mathcal{L}$ be a leaf node and $N^{\mathcal{L}}$ the total number of instances in $\mathcal{L}$. For each label y_j, let $n^{\mathcal{L}}(y_j)$ ($n^{\mathcal{L}}(\overline{y_j})$) denote the number of instances in $\mathcal{L}$ for which y_j is relevant (irrelevant). Let us consider the A-NPI-M probability interval on y_j on $\mathcal{L}$:

$$I^{\mathcal{L}}_{ANPI}(y_j) = \left[\max\left(\frac{n^{\mathcal{L}}(y_j) - 1}{N^{\mathcal{L}}}, 0\right), \min\left(\frac{n^{\mathcal{L}}(y_j) + 1}{N^{\mathcal{L}}}, 1\right)\right]. \tag{11.7}$$

The following credal set is associated with this interval:

$$\mathcal{P}\left(I^{\mathcal{L}}_{ANPI}(y_j)\right) = \left\{p \in \mathcal{P}(y_j) \mid p(y_j) \in I^{\mathcal{L}}_{ANPI}(y_j)\right\}. \tag{11.8}$$

Let $\hat{p}^{\mathcal{L}}_j(y_j)$ be the probability distribution of maximum entropy on $\mathcal{P}\left(I^{\mathcal{L}}_{ANPI}(y_j)\right)$. It can be directly computed by means of Algorithm 14, and is determined as follows:

$$\hat{p}^{\mathcal{L}}_j(y_j) = \begin{cases} \frac{1}{2} & \text{if} \quad \left|n^{\mathcal{L}}(y_j) - n^{\mathcal{L}}(\overline{y_j})\right| \leqslant 2 \\ \frac{n^{\mathcal{L}}(y_j) - 1}{N^{\mathcal{L}}} & \text{if} \quad n^{\mathcal{L}}(y_j) > n^{\mathcal{L}}(\overline{y_j}) + 2 \\ \frac{n^{\mathcal{L}}(y_j) + 1}{N^{\mathcal{L}}} & \text{if} \quad n^{\mathcal{L}}(\overline{y_j}) > n^{\mathcal{L}}(y_j) + 2 \end{cases} \tag{11.9}$$

The set of labels predicted as relevant by ML-CDT at $\mathcal{L}$ is composed of those labels for which the predicted posterior probability is greater or equal than 0.5:

$$h^{\mathcal{L}}_{ML\_CDT} = \left\{y_j, 1 \leqslant j \leqslant n_L \mid \hat{p}^{\mathcal{L}}_j(y_j) \geqslant 0.5\right\}. \tag{11.10}$$

Algorithm 22 summarizes the building procedure of a ML-CDT.

In order to classify an instance with ML-CDT, a path from the root node to a leaf one is made by using the attribute values of the instance. The predicted label set for the instance is the one assigned to such a terminal node. The same occurs with the predicted posterior probabilities about the relevance of

Algorithm 22: Procedure to build a Multi-Label Credal Decision Tree.

```
Procedure Build_ML-CDT(Node $\mathcal{N}$)
Let $\mathcal{D}$ be the dataset associated with $\mathcal{N}$
if there are more attributes to insert then
    Select $X^i$ the attribute that reaches the maximum value of
      $IIG^{\mathcal{D}}(\mathcal{Y}, X^i)$
    for $x^i_{r_i}$ possible value of $X^i$ do
        Make a node $\mathcal{N}_{r_i}$ child of $\mathcal{N}$
        Build_ML-CDT($\mathcal{N}_{r_i}$)
else
    Make $\mathcal{N}$ a leaf node
    for $j = 1$ to $n_L$ do
        $f^{\mathcal{N}}_{ML\_CDT}(y_j) = \hat{p}^{\mathcal{L}}_j(y_j)$,
        where $\hat{p}^{\mathcal{L}}_j(y_j)$ is determined by Equation (11.9)
    Assign a label set $h^{\mathcal{N}}_{ML\_CDT}$ to $\mathcal{N}$, computed through Equation
      (11.10)
```

Algorithm 23: Procedure to classify an instance with ML-CDT.

```
Procedure Classify_ML-CDT(ML-CDT $\mathcal{T}$, instance with attribute vector
$\mathbf{x} = (x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d})$)
1. Follow a path in $\mathcal{T}$ from the root node to a leaf one $\mathcal{L}$ using the
   attribute values $x^1_{r_1}, x^2_{r_2}, \dots, x^d_{r_d}$.
2. for $j = 1$ to $n_L$ do
    $f^{ML\_CDT}(\mathbf{x}, y_j) = f^{\mathcal{L}}_{ML\_CDT}(y_j)$
3. Assign the predicted label set at $\mathcal{L}$, $h^{\mathcal{L}}_{ML\_CDT}$, to $h^{ML\_CDT}(\mathbf{x})$.
```

the labels for such an instance. Algorithm 23 summarizes the procedure to classify an instance via ML-CDT.

For handling continuous attributes, similar to ML-DT, ML-CDT considers binary splits and selects the split point that produces the maximum IIG value. Concerning missing values, when an instance has a missing value for an attribute, it can go down each branch hanging from the corresponding node with a weight equal to the proportion of instances at such a branch, as in ML-DT. Similar to ML-DT, ML-CDT can utilize pruning processes by considering the number of errors in all labels.

11.3.1 Differences between Multi-Label Decision Tree and Multi-Label Credal Decision Tree

We remark below the main differences between the behavior of ML-DT and ML-CDT.

- **Size of the dataset**: It is easy to observe that the intervals $I^{\mathcal{D}}_{ANPI}(y_j)$, determined through Equation (11.1), are narrower as the number of instances in the dataset, $N^{\mathcal{D}}$, is higher. In consequence, as $N^{\mathcal{D}}$ is higher, the corresponding credal set, computed via Equation (11.2), has fewer probability distributions far from the one associated with relative frequencies. Hence, at the upper levels of the tree, where the number of instances tends to be pretty large, IG and IIG, defined in Equations (6.53) and (11.6), respectively, may provide similar values and, thus, ML-DT and ML-CDT might have similar behavior. In contrast, at the lower levels of the tree, where there are often very few instances, the associated credal sets may contain many probability distributions far from the ones estimated through relative frequencies. So, in these cases, IG and IIG might not give similar values since $S(\mathcal{Y})$ and $S^*(\mathcal{Y})$ are probably quite different. In this way, ML-DT and ML-CDT behave similarly at the upper levels of the tree but their behavior might be quite different at the lower levels. The same happens with classical Decision Trees and Credal Decision Trees for traditional classification, as argued in Section 4.6.3.

- **Stop criterion**: For an attribute X^i that takes values in $\{x^i_1, x^i_2, \ldots, x^i_{t_i}\}$, the value of $IIG(\mathcal{Y}, X^i)$ can be negative, unlike $IG(\mathcal{Y}, X^i)$. The reason is that, according to the results proved in [5], the imprecise information gain for a label y_j, $S^*\left(\mathcal{P}\left(I^{\mathcal{D}}(y_j)\right)\right) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) S^*\left(\mathcal{Y} \mid X^i = x^i_{r_i}\right)$, can be negative, unlike the information gain $S^{\mathcal{D}}(\mathcal{Y}) - \sum_{r_i=1}^{t_i} P^{\mathcal{D}}(X^i = x^i_{r_i}) S^{\mathcal{D}}(\mathcal{Y} \mid X^i = x^i_{r_i})$. Therefore, ML-CDT avoids selecting attributes that worsen the uncertainty-based information about the label set. In consequence, overfitting in ML-CDT is probably lower than in ML-DT because ML-CDT may stop branching the tree before ML-DT.

- **Predictions at leaf nodes**: We may note that ML-DT predicts a label y_j as relevant at a leaf node $\mathcal{L}$ if, and only if, $n^{\mathcal{L}}(y_j) \geqslant n^{\mathcal{L}}(\overline{y_j})$, $n^{\mathcal{L}}(y_j)$ and $n^{\mathcal{L}}(\overline{y_j})$ being the number of instances in $\mathcal{L}$ for which y_j is relevant and irrelevant, respectively. Nonetheless, ML-CDT predicts y_j as relevant at $\mathcal{L} \Leftrightarrow n^{\mathcal{L}}(y_j) + 2 \geqslant n^{\mathcal{L}}(\overline{y_j})$. Thus, ML-CDT is more flexible than ML-DT for predicting a label as relevant at a leaf node. It is suitable for alleviating the class-imbalance problem that usually appears in MLC.

  Regarding the predicted posterior probabilities about the relevance of the labels for the instances, ML-DT predicts the ones associated with relative frequencies, while ML-CDT predicts the probability distributions that yield the maximum entropies on the corresponding credal sets. Consequently, according to the examples shown below, the posterior probabilities predicted by ML-CDT might be less sensitive to label noise than the posterior probabilities predicted by ML-DT.

- **Noise in the labels**: We show with an example below that, for a certain label, the Shannon entropy may be more sensitive to noise in that label than the maximum entropy on the corresponding credal set:

  Proposition 11.3.1 *Suppose that there is a dataset $\mathcal{D}$ with $N^{\mathcal{D}}$ instances. Let $n^{\mathcal{D}}(y_j)$ denote the number of instances in $\mathcal{D}$ that have associated y_j and $n^{\mathcal{D}}(\overline{y_j})$ the number of instances in $\mathcal{D}$ for which y_j is irrelevant. Let us assume that $n^{\mathcal{D}}(\overline{y_j}) > n^{\mathcal{D}}(y_j) + 3$ and $n^{\mathcal{D}}(y_j) > 2$. Let $\mathcal{D}_{nois}$ be a dataset derived from $\mathcal{D}$ by chaining the value of y_j for an instance from irrelevant to relevant. With these assumptions, it holds that*

$$S^{\mathcal{D}_{nois}}(y_j) - S^{\mathcal{D}}(y_j) \geqslant S^{*}\left(\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)\right) - S^{*}\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right),$$

  where $\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)$ $\left(\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)\right)$ is the A-NPI-M credal set on y_j corresponding to $\mathcal{D}$ ($\mathcal{D}_{nois}$), determined by Equation (11.2).

  Proof: We have that:

$$S^{\mathcal{D}}(y_j) = -\frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}}\log_2\frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}} - \frac{n^{\mathcal{D}}(\overline{y_j})}{N^{\mathcal{D}}}\log_2\frac{n^{\mathcal{D}}(\overline{y_j})}{N^{\mathcal{D}}},$$

$$S^{\mathcal{D}_{nois}}(y_j) = -\frac{n^{\mathcal{D}}(\overline{y_j})-1}{N^{\mathcal{D}}}\log_2\frac{n^{\mathcal{D}}(\overline{y_j})-1}{N^{\mathcal{D}}} - \frac{n^{\mathcal{D}}(y_j)+1}{N^{\mathcal{D}}}\log_2\frac{n^{\mathcal{D}}(y_j)+1}{N^{\mathcal{D}}}.$$

Hence,

$$\begin{aligned}
&S^{\mathcal{D}_{nois}}(y_j) - S^{\mathcal{D}}(y_j) = -\frac{n^{\mathcal{D}}(\overline{y_j}) - 1}{N^{\mathcal{D}}} \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) - \\
&\frac{n^{\mathcal{D}}(y_j) + 1}{N^{\mathcal{D}}} \log_2(n^{\mathcal{D}}(y_j) + 1) + \frac{n^{\mathcal{D}}(\overline{y_j})}{N^{\mathcal{D}}} \log_2(n^{\mathcal{D}}(\overline{y_j})) + \frac{n^{\mathcal{D}}(y_j)}{N^{\mathcal{D}}} \log_2(n^{\mathcal{D}}(y_j)) + \\
&\frac{\log_2(N^{\mathcal{D}})}{N^{\mathcal{D}}} \times \left[n^{\mathcal{D}}(\overline{y_j}) - 1 + n^{\mathcal{D}}(y_j) + 1 - n^{\mathcal{D}}(\overline{y_j}) - n^{\mathcal{D}}(y_j) \right] = \\
&\frac{-(n^{\mathcal{D}}(\overline{y_j}) - 1) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) - (n^{\mathcal{D}}(y_j) + 1) \log_2(n^{\mathcal{D}}(y_j) + 1)}{N^{\mathcal{D}}} + \\
&\frac{n^{\mathcal{D}}(\overline{y_j}) \log_2(n^{\mathcal{D}}(\overline{y_j})) + n^{\mathcal{D}}(y_j) \log_2(n^{\mathcal{D}}(y_j))}{N^{\mathcal{D}}}.
\end{aligned}$$

Now, we may note that the probability distribution that reaches the maximum entropy on $\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)$ is given by $\hat{p}_j^{\mathcal{D}}(y_j) = \frac{n^{\mathcal{D}}(y_j)+1}{N^{\mathcal{D}}}$. Likewise, it is easy to check that the probability distribution of maximum entropy on $\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)$ is determined by $\hat{p}_j^{\mathcal{D}_{nois}}(y_j) = \frac{n^{\mathcal{D}}(y_j)+2}{N^{\mathcal{D}}}$.

Consequently,

$$\begin{aligned}
&S^*\left(\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)\right) - S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) = \\
&\frac{-(n^{\mathcal{D}}(\overline{y_j}) - 2) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 2) - (n^{\mathcal{D}}(y_j) + 2) \log_2(n^{\mathcal{D}}(y_j) + 2)}{N^{\mathcal{D}}} + \\
&\frac{(n^{\mathcal{D}}(\overline{y_j}) - 1) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) + (n^{\mathcal{D}}(y_j) + 1) \log_2(n^{\mathcal{D}}(y_j) + 1)}{N^{\mathcal{D}}}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
&S^{\mathcal{D}_{nois}}(y_j) - S^{\mathcal{D}}(y_j) \geqslant S^*\left(\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)\right) - S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) \Leftrightarrow \\
&-(n^{\mathcal{D}}(\overline{y_j}) - 1) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) - (n^{\mathcal{D}}(y_j) + 1) \log_2(n^{\mathcal{D}}(y_j) + 1) + \\
&n^{\mathcal{D}}(\overline{y_j}) \log_2(n^{\mathcal{D}}(\overline{y_j})) + n^{\mathcal{D}}(y_j) \log_2(n^{\mathcal{D}}(y_j)) \geqslant \\
&-(n^{\mathcal{D}}(\overline{y_j}) - 2) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 2) - (n^{\mathcal{D}}(y_j) + 2) \log_2(n^{\mathcal{D}}(y_j) + 2) + \\
&(n^{\mathcal{D}}(\overline{y_j}) - 1) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) + (n^{\mathcal{D}}(y_j) + 1) \log_2(n^{\mathcal{D}}(y_j) + 1) \Leftrightarrow \\
&n^{\mathcal{D}}(\overline{y_j}) \log_2(n^{\mathcal{D}}(\overline{y_j})) + n^{\mathcal{D}}(y_j) \log_2(n^{\mathcal{D}}(y_j)) + \\
&(n^{\mathcal{D}}(\overline{y_j}) - 2) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 2) + (n^{\mathcal{D}}(y_j) + 2) \log_2(n^{\mathcal{D}}(y_j) + 2) \geqslant \\
&2(n^{\mathcal{D}}(\overline{y_j}) - 1) \log_2(n^{\mathcal{D}}(\overline{y_j}) - 1) + 2(n^{\mathcal{D}}(y_j) + 1) \log_2(n^{\mathcal{D}}(y_j) + 1)
\end{aligned}$$

As the logarithm function is convex, it holds that:

$$\begin{aligned}
&n^{\mathcal{D}}(\overline{y_j})\log_2(n^{\mathcal{D}}(\overline{y_j})) + (n^{\mathcal{D}}(\overline{y_j}) - 2)\log_2(n^{\mathcal{D}}(\overline{y_j}) - 2) \geqslant \\
&2(n^{\mathcal{D}}(\overline{y_j}) - 1)\log_2(n^{\mathcal{D}}(\overline{y_j}) - 1), \\
&(n^{\mathcal{D}}(y_j) + 2)\log_2(n^{\mathcal{D}}(y_j) + 2) + n^{\mathcal{D}}(y_j)\log_2(n^{\mathcal{D}}(y_j)) \geqslant \\
&2(n^{\mathcal{D}}(y_j) + 1)\log_2(n^{\mathcal{D}}(y_j) + 1),
\end{aligned}$$

and it is quite easy to check that our thesis holds.

□

Remark that the main difference between the split criteria of ML-CDT and ML-DT is that the former is based on the maximum entropies on the A-NPI-M credal sets corresponding to the labels, whereas the latter is based on the Shannon entropy of each label. Hence, the previous proposition lets us deduce that, when noise is introduced in a dataset by changing the value of a label for an instance, the split criterion used in ML-CDT may be less sensitive to such a change than the one employed in ML-DT. We show below an example of this point, which is very based on the one given in [150].

Example 11.3.1 *Let $\mathcal{D}$ be a dataset of $N^{\mathcal{D}} = 15$ instances. Suppose that a label y_j is irrelevant for 5 instances and that the other 10 instances have associated y_j. Let $n^{\mathcal{D}}(y_j)$ and $n^{\mathcal{D}}(\overline{y_j})$ denote the number of instances in $\mathcal{D}$ for which y_j is relevant and irrelevant, respectively. Let X^1 and X^2 be two binary attributes. Let us assume the following arrangement for each one of the possible values of the attributes:*

$$\begin{aligned}
X^1 = 0 &\rightarrow (n^{\mathcal{D}}(y_j) = 4, n^{\mathcal{D}}(\overline{y_j}) = 5), \\
X^1 = 1 &\rightarrow (n^{\mathcal{D}}(y_j) = 6, n^{\mathcal{D}}(\overline{y_j}) = 0), \\
X^2 = 0 &\rightarrow (n^{\mathcal{D}}(y_j) = 1, n^{\mathcal{D}}(\overline{y_j}) = 5), \\
X^2 = 1 &\rightarrow (n^{\mathcal{D}}(\overline{y_j}) = 9, n^{\mathcal{D}}(\overline{y_j}) = 0).
\end{aligned}$$

We have that:

$$S^{\mathcal{D}}(y_j) = -\frac{10}{15}\log_2\left(\frac{10}{15}\right) - \frac{5}{15}\log_2\left(\frac{5}{15}\right) = 0.918,$$

$$S^{\mathcal{D}}(y_j \mid X^1 = 0) = -\frac{4}{9}\log_2\left(\frac{4}{9}\right) - \frac{5}{9}\log_2\left(\frac{5}{9}\right) = 0.991, \quad S^{\mathcal{D}}(y_j \mid X^1 = 1) = 0,$$

$$\begin{aligned}IG^{\mathcal{D}}(y_j, X^1) = &S^{\mathcal{D}}(y_j) - P^{\mathcal{D}}(X^1=0)S^{\mathcal{D}}(y_j \mid X^1=0) - \\ &P^{\mathcal{D}}(X^1=1)S^{\mathcal{D}}(y_j \mid X^1=1) \\ &= 0.918 - 0.991 \times 0.6 = 0.3237.\end{aligned}$$

Regarding X^2*:*

$$S^{\mathcal{D}}(y_j \mid X^2=0) = -\frac{1}{6}\log_2\left(\frac{1}{6}\right) - \frac{5}{6}\log_2\left(\frac{5}{6}\right) = 0.65, \quad S^{\mathcal{D}}(y_j \mid X^2=1) = 0,$$

$$\begin{aligned}IG^{\mathcal{D}}(y_j, X^2) = &S^{\mathcal{D}}(y_j) - P^{\mathcal{D}}(X^2=0)S^{\mathcal{D}}(y_j \mid X^2=0) \\ &- P^{\mathcal{D}}(X^2=1)S^{\mathcal{D}}(y_j \mid X^2=1) \\ &= 0.918 - 0.65 \times 0.4 = 0.6583.\end{aligned}$$

Let $\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)$ *denote the A-NPI-M credal set on* y_j *associated with* $\mathcal{D}$*, computed by means of Equation (11.2). It holds that:*

$$\begin{aligned}&S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) = 0.971, \\ &S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^1=0\right)\right) = 1, \quad S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^1=1\right)\right) = 0.65,\end{aligned}$$

$$\begin{aligned}IIG^{\mathcal{D}}(y_j, X^1) = &S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) \\ &- P^{\mathcal{D}}(X^1=0)S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^1=0\right)\right) \\ &- P^{\mathcal{D}}(X^1=1)S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^1=1\right)\right) \\ &= 0.971 - 0.6 - 0.26 \times 0.4 = 0.111.\end{aligned}$$

For X^2*:*

$$\begin{aligned}&S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^2=0\right)\right) = 0.65, \\ &S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^2=1\right)\right) = 0.5033.\end{aligned}$$

$$\begin{aligned}IIG^{\mathcal{D}}(y_j, X^2) = &S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j)\right)\right) \\ &- P^{\mathcal{D}}(X^2=0)S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^2=0\right)\right) \\ &- P^{\mathcal{D}}(X^2=1)S^*\left(\mathcal{P}\left(I^{\mathcal{D}}_{ANPI}(y_j) \mid X^2=1\right)\right) \\ &= 0.971 - 0.4 \times 0.65 - 0.5033 \times 0.6 = 0.409.\end{aligned}$$

As $\mathrm{IG}^{\mathcal{D}}(y_j, X^1) < \mathrm{IG}^{\mathcal{D}}(y_j, X^2)$ *and* $\mathrm{IIG}^{\mathcal{D}}(y_j, X^1) < \mathrm{IIG}^{\mathcal{D}}(y_j, X^2)$*, if we only had the label* y_j*, the attribute* X^2 *would be selected to branch the tree in both ML-DT and ML-CDT.*

Suppose that noise is introduced in the dataset by changing the value of y_j *for an instance that verifies that* $X^1 = 0$ *and* $X^2 = 1$ *from relevant to irrelevant. In this noisy dataset,* $\mathcal{D}_{nois}$*, the instances are arranged as follows:*

$$\begin{aligned}
X^1 &= 0 \to (n^{\mathcal{D}_{nois}}(y_j) = 3, n^{\mathcal{D}_{nois}}(\overline{y_j}) = 6),\\
X^1 &= 1 \to (n^{\mathcal{D}_{nois}}(y_j) = 6, n^{\mathcal{D}_{nois}}(\overline{y_j}) = 0),\\
X^2 &= 0 \to (n^{\mathcal{D}_{nois}}(y_j) = 1, n^{\mathcal{D}_{nois}}(\overline{y_j}) = 5),\\
X_2 &= 1 \to (n^{\mathcal{D}_{nois}}(y_j) = 8, n^{\mathcal{D}_{nois}}(\overline{y_j}) = 1),
\end{aligned}$$

where $n^{\mathcal{D}_{nois}}(y_j)$ $\left(n^{\mathcal{D}_{nois}}(\overline{y_j})\right)$ *denotes the number of instances in* $\mathcal{D}_{nois}$ *for which* y_j *is relevant (irrelevant).*

Let $\mathcal{P}\left(I^{\mathcal{D}_{nois}}_{ANPI}(y_j)\right)$ *denote the A-NPI-M credal set on* y_j *corresponding to* $\mathcal{D}_{nois}$*, computed by means of Equation (11.2). Values of* IG *and* IIG *in this noisy dataset:*

$$\begin{aligned}
S^{\mathcal{D}_{nois}}(y_j) &= -\frac{9}{15}\log_2\left(\frac{9}{15}\right) - \frac{6}{15}\log_2\left(\frac{6}{15}\right) = 0.971,\\
S^{\mathcal{D}_{nois}}(y_j \mid X^1 = 0) &= -\frac{3}{9}\log_2\left(\frac{3}{9}\right) - \frac{6}{9}\log_2\left(\frac{6}{9}\right) = 0.9183,\\
S^{\mathcal{D}_{nois}}(y_j \mid X^1 = 1) &= 0,
\end{aligned}$$

$$\begin{aligned}
\mathrm{IG}^{\mathcal{D}_{nois}}(y_j, X^1) &= S^{\mathcal{D}_{nois}}(y_j) - P^{\mathcal{D}_{nois}}(X^1 = 0)S^{\mathcal{D}_{nois}}(y_j \mid X^1 = 0)\\
&\quad - P^{\mathcal{D}_{nois}}(X^1 = 1)S^{\mathcal{D}_{nois}}(y_j \mid X^1 = 1)\\
&= 0.971 - 0.9183 \times 0.6 = 0.42.
\end{aligned}$$

$$\begin{aligned}
S^{\mathcal{D}_{nois}}(y_j \mid X^2 = 0) &= -\frac{1}{6}\log_2\left(\frac{1}{6}\right) - \frac{5}{6}\log_2\left(\frac{5}{6}\right) = 0.65,\\
S^{\mathcal{D}_{nois}}(y_j \mid X^2 = 1) &= -\frac{8}{9}\log_2\left(\frac{8}{9}\right) - \frac{1}{9}\log_2\left(\frac{1}{9}\right) = 0.5033,
\end{aligned}$$

$$\begin{aligned}
\mathrm{IG}^{\mathcal{D}_{nois}}(y_j, X^2) &= S^{\mathcal{D}_{nois}}(y_j) - P^{\mathcal{D}_{nois}}(X^2 = 0)S^{\mathcal{D}_{nois}}(y_j \mid X^2 = 0)\\
&\quad - P^{\mathcal{D}_{nois}}(X^2 = 1)S^{\mathcal{D}_{nois}}(y_j \mid X^2 = 1)\\
&= 0.971 - 0.65 \times 0.4 - 0.5033 \times 0.6 = 0.409.
\end{aligned}$$

$$S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j)\right)\right) = 0.9968,$$
$$S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^1 = 0)\right)\right) = 0.9911, \quad S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^1 = 1)\right)\right) = 0.65,$$

$$\begin{aligned} IIG^{\mathcal{D}_{nois}}(y_j, X^1) = S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j)\right)\right) \\ - P^{\mathcal{D}_{nois}}(X^1 = 0) S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^1 = 0)\right)\right) \\ - P^{\mathcal{D}_{nois}}(X^1 = 1) S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^1 = 1)\right)\right) \\ = 0.9968 - 0.6 \times 0.9911 - 0.65 \times 0.4 = 0.1421. \end{aligned}$$

$$S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^2 = 0)\right)\right) = 0.65,$$
$$S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^2 = 1)\right)\right) = 0.5033,$$

$$\begin{aligned} IIG^{\mathcal{D}_{nois}}(y_j, X^2) = S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j)\right)\right) \\ - P^{\mathcal{D}_{nois}}(X^2 = 0) S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^2 = 0)\right)\right) \\ - P^{\mathcal{D}_{nois}}(X^2 = 1) S^*\left(\mathcal{P}\left(I_{ANPI}^{\mathcal{D}_{nois}}(y_j \mid X^2 = 1)\right)\right) \\ = 0.9968 - 0.4 \times 0.65 - 0.5033 \times 0.6 = 0.4055. \end{aligned}$$

In this noisy dataset, it holds that $IG^{\mathcal{D}_{nois}}(y_j, X^2) < IG^{\mathcal{D}_{nois}}(y_j, X^1)$ *and* $IIG^{\mathcal{D}_{nois}}(y_j, X^1) < IIG^{\mathcal{D}_{nois}}(y_j, X^2)$. *Thereby, in ML-DT, the attribute* X^1 *is now selected for splitting the dataset (assuming that* y_j *is the only label). Nevertheless, in ML-CDT, the selected split attribute is* X^2*, as with the clean dataset.*

Hence, in the previous example, IIG is not affected by the noise in the label y_j, unlike IG. It illustrates the fact that the split criterion employed in ML-CDT is probably less sensitive to label noise than the one used in ML-DT. Therefore, it can be deduced that ML-CDT might be more robust to label noise than ML-DT since the main difference between both algorithms resides in the split criterion.

In summary, ML-CDT may be less sensitive to label noise than ML-DT. Moreover, we must take into account that the intrinsic label noise in MLC might be higher than in traditional classification, as argued in Section 11.2.3. For this reason, it is expected that ML-CDT performs better than ML-DT. This issue is checked with an experimental analysis carried out in Section 11.3.2.

11.3.2 Experiments

In this experimental study, we aim to compare the performance of the existing ML-DT and our proposed ML-CDT.

11.3.2.1 *Experimental settings*

- **Datasets**: For our experimentation, we have employed the datasets used in the experimental analysis of the previous section, except for 'delicious'. We have not used this dataset because it requires a very high computational cost due to its high number of labels and instances. Table 11.1 shows the main characteristics of these datasets: number of instances, number of continuous and discrete attributes, number of labels, label cardinality, label density, and MLC domain.

- **Algorithms**: Two MLC algorithms are compared in our experimental study: ML-CDT and ML-DT.[3]

- **Evaluation metrics**: In accordance with the extensive experimental analysis with MLC methods carried out in [144], sixteen evaluation metrics have been employed in this experimentation: Six are based on instance-classification: Hamming Loss, Subset Accuracy, Accuracy, Precision, Recall, and F1; six label-based classification evaluation measures have been employed: Micro Precision, Macro Precision, Micro Recall, Macro Recall, Micro F1, and Macro F1; the other four evaluation metrics are based on ranking: Coverage, One Error, Ranking Loss, and Average Precision. We detailed all these metrics in Section 6.3.

- **Procedure**: Three noise levels have been considered in this experimental study: 0%, 5%, and 10%. For each algorithm, dataset, and noise level, the following cross-validation procedure has been carried out: the dataset is divided into five partitions, and, for each one of them, an iteration has been done. In it, the corresponding partition is utilized for testing, and the rest of the data for training. For each label, the x% of the training instances (x being the noise level) are chosen, and the value of their label is changed (if the label is irrelevant it is changed to relevant and vice-

[3] In this experimentation, a version of the ML-CDT algorithm that uses uncertainty measures on IDM credal sets has not been considered because the IDM has a strong dependence on a parameter. In fact, with the standard value of the IDM parameter, the obtained results are poor.

versa)[4]. The MLC model is learned with the noisy training set, and the evaluation measures are extracted via the test set. The same partitions have been employed for ML-DT and ML-CDT in all datasets.

- **Software**: We have implemented both algorithms in Mulan. For this purpose, we have used some of the structures provided in Weka. We have used part of the functionality available in Mulan to create the partitions of cross-validation. In order to generate the label noise, we have utilized the Weka filters.

- **Statistical evaluation**: For each evaluation metric and noise level considered here, we have two algorithms to compare: ML-DT and ML-CDT. In this way, consistently the indications given in [49, 75] for statistical comparisons between two methods, the Wilcoxon test has been used with a level of significance of $\alpha = 0.05$ to check which algorithm performs better and whether the differences are statistically significant.

11.3.2.2 *Results and discussion*

Table 11.4 shows a summary of the results obtained by ML-DT and ML-CDT for each metric and noise level. Specifically, for each metric and noise level, it shows which method performs better according to the Wilcoxon test and whether the differences are statistically significant. Also, Table 11.4 lets us see, for each metric and noise level, in how many datasets an algorithm achieves a better result than the other one.

We must express the following comments about these results for each type of evaluation measure:

Instance-based classification metrics:

- ML-CDT significantly outperforms ML-DT through the Wilcoxon test in Hamming Loss for all noise levels. In addition, for the three noise levels, the number of datasets in which ML-CDT achieves a better result than ML-DT in Hamming Loss is far higher than the number of datasets where the contrary happens. Consequently, there are fewer pairs of label-instance erroneously classified with ML-CDT than with ML-DT.

[4] We have not considered noise levels higher than 10% because the noise is introduced in each label and, thus, a higher noise level would imply a very considerable amount of noise in the data.

Table 11.4: Summary of the results obtained by ML-DT and ML-CDT for each evaluation metric and label noise level. (•) indicates that the algorithm of the column performs significantly better than the other one according to the Wilcoxon test. (-) means that the classifier of the column performs better than the other one but the results are statistically equivalent.

Noise level	Metric	ML-DT	ML-CDT	Wins ML-DT	Wins ML-CDT
0% Noise	Hamming Loss		(•)	0	12
	Subset Accuracy		(-)	2	9
	Accuracy	(-)		5	7
	Precision		(-)	2	10
	Recall	(•)		11	1
	F1	(-)		5	7
	Micro Precision		(•)	0	12
	Macro Precision		(-)	5	7
	Micro Recall	(•)		11	1
	Macro Recall	(•)		10	2
	Micro F1	(-)		5	7
	Macro F1		(-)	4	8
	Coverage		(•)	1	11
	Ranking Loss		(•)	1	11
	Average Precision		(-)	3	9
	One Error		(-)	3	9
5% Noise	Hamming Loss		(•)	0	12
	Subset Accuracy		(•)	2	9
	Accuracy		(-)	3	9
	Precision		(•)	1	11
	Recall	(•)		10	2
	F1		(-)	3	9
	Micro Precision		(•)	1	10
	Macro Precision		(-)	2	10
	Micro Recall	(•)		12	0
	Macro Recall	(•)		12	0
	Micro F1		(-)	3	9
	Macro F1	(-)		7	5
	Coverage		(•)	2	10
	Ranking Loss		(•)	2	10
	Average Precision		(-)	3	9
	One Error		(-)	3	9
10% Noise	Hamming Loss		(•)	0	12
	Subset Accuracy		(•)	0	11
	Accuracy		(•)	2	10
	Precision		(•)	1	11
	Recall	(•)		11	1
	F1		(•)	2	10
	Micro Precision		(•)	1	11
	Macro Precision		(•)	1	11
	Micro Recall	(•)		11	1
	Macro Recall	(•)		11	1
	Micro F1		(•)	2	10
	Macro F1	(-)		6	6
	Coverage		(•)	2	10
	Ranking Loss		(•)	2	10
	Average Precision		(-)	4	8
	One Error		(-)	4	8

- The results obtained in Recall (Wilcoxon test and number of wins of each algorithm for each noise level) allows deducing that ML-DT predicts more relevant labels as relevant than ML-CDT.

- When there is no noise in the labels, the results obtained by ML-DT and ML-CDT in the rest of the instance-based classification metrics are statistically equivalent according to the Wilcoxon test. However, the number of wins of ML-CDT in Precision and Subset Accuracy is considerably higher than the wins of ML-DT in these metrics. Hence, ML-CDT predicts fewer irrelevant labels as relevant than ML-DT (Precision), and the former algorithm correctly predicts the entire set of relevant labels for more instances than the latter (Subset Accuracy).

- With a 5% of label noise, ML-CDT performs significantly better than ML-DT in Precision and Subset Accuracy according to the Wilcoxon test. The results obtained by ML-DT and ML-CDT in F1 (harmonic mean between Precision and Recall) and Accuracy, which measures how an algorithm predicts the label sets for the instances in general, are statistically equivalent according to the Wilcoxon test. Nevertheless, in these metrics, the number of wins of ML-CDT is notably higher than the number of wins of ML-DT (9 versus 3 in both metrics).

- When there is a 10% of noise in the labels, ML-CDT obtains significantly better performance than ML-DT in all instance-based classification metrics, except for Recall. In this measure, the results are significantly better for ML-DT according to the Wilcoxon test.

Label-based classification measures:

- For all noise levels, according to the Wilcoxon test, ML-CDT obtains significantly better results than ML-DT in the Precision averaged overall pairs of label-instance (Micro Precision). In contrast, ML-DT always significantly outperforms ML-CDT via the Wilcoxon test in the Recall averaged over all labels (Macro Recall) and in the Recall averaged overall pairs of label-instance (Micro Recall).

- When there is no noise in the labels, the results obtained by ML-DT and ML-CDT in the remaining label-based classification metrics are statistically equivalent according to the Wilcoxon test.

- The results are similar when there is a 5% of noise in the labels. However, we should remark that, in Macro Precision, which is the Precision averaged overall labels, ML-CDT achieves a better result than ML-DT in ten datasets, whereas ML-DT outperforms ML-CDT in only two datasets.

- With a 10% of label noise, ML-CDT obtains significantly better results than ML-DT according to the Wilcoxon test in Macro Precision. Also, ML-CDT significantly outperforms ML-DT via the Wilcoxon test in terms of the harmonic mean between Micro Precision and Micro Recall (Micro F1). The results obtained by ML-DT and ML-CDT in Macro F1, the harmonic mean between Precision and Recall averaged overall labels, are statistically equivalent according to the Wilcoxon test, as happens with 0% and 5% of noise.

Ranking-based metrics:

- In general, the results obtained by ML-CDT in the ranking-based metrics are better than the ones obtained by ML-DT. This issue can be appreciated in the number of datasets in which an algorithm outperforms the other and the results of the Wilcoxon tests. In consequence, ML-CDT generally predicts the posterior probabilities about the relevance of the labels for the instances better than ML-DT.

- Specifically, according to the Wilcoxon test, ML-CDT performs significantly better than ML-DT in Coverage, which indicates the average number of steps that are necessary to go down the predicted label ranking for covering all relevant labels for an instance, and in Ranking Loss (the average number of pairs of relevant/irrelevant labels reversely ordered).

- For all label noise levels, in the other two ranking-based evaluation measures, the results obtained by ML-DT and ML-CDT are statistically equivalent according to the Wilcoxon test. Nonetheless, in both measures, the number of wins of ML-CDT is notably higher than the number of wins of ML-DT for all noise levels.

Summary of the results: We can summarize the results obtained in this experimentation in the following points:

- Due to the results obtained in Accuracy, Hamming Loss, and Subset Accuracy, we can state that ML-CDT generally predicts the sets of relevant labels for the instances better than ML-DT. This issue is enhanced as the noise in the labels is higher. The reasons are that, as argued in Section 11.2.3, the intrinsic label noise in MLC may be higher than in traditional classification, and ML-CDT is probably more robust to label noise than ML-DT, as shown in Section 11.3.1.

- ML-DT always performs significantly better than ML-CDT in the metrics associated with Recall. It means that ML-DT predicts more relevant labels as relevant than ML-CDT. In contrast, ML-CDT predicts fewer irrelevant labels as relevant than ML-DT due to the better results in the measures corresponding to Precision. The performance of both algorithms in the metrics associated with F1 (harmonic mean between Precision and Recall) is statistically equivalent without noise. However, when there is noise in the labels, the results are more favorable to ML-CDT. These facts happen because, in MLC, there are normally very few instances that have associated a certain label, and ML-CDT might stop branching the tree before ML-DT. Thus, in many cases, ML-CDT does not reach parts of the tree where relevant labels are correctly predicted. Nevertheless, in these parts of the tree, the noise has a quite negative influence, and, there, irrelevant labels are sometimes erroneously predicted as relevant. Since ML-CDT might be more appropriate than ML-DT for handling label noise, as the label noise is higher, the harmonic means between Precision and Recall are more favorable to ML-CDT.

- For all noise levels, ML-CDT predicts more suitable posterior probabilities about the relevance of the labels for the instances than ML-DT. This occurs because the noise has a negative influence on the label rankings predicted by the adaptations of Decision Trees for MLC, and the building process of ML-CDT may be more robust to label noise than the building process of ML-DT. Moreover, the posterior probabilities predicted by ML-CDT at a leaf node are the ones that attain the maximum entropy on the A-NPI-M credal sets on such a leaf node, while the posterior probabilities predicted by ML-DT are the ones estimated by relative frequencies at the terminal node. Hence, the predicted posterior probabilities are less sensitive to label noise in the case of ML-CDT. In addition, at leaf nodes, there are often very few instances. Thus, at leaf nodes, the difference between the behavior of ML-DT and ML-CDT is even more notable.

- Therefore, it can be stated that, as expected, ML-CDT performs better than ML-DT, the improvement being more notable as there is more noise in the labels.

11.4 Lazy Multi-Label Classification methods based on imprecise probabilities

11.4.1 Multi-Label Credal Nearest Neighbors

The Multi-Label Nearest Neighbors algorithm (ML-Credal-NN), presented in this section, uses a Maximum a Posteriori principle (MAP) to predict the label set associated with an instance that is required to be classified. Similar to ML-NN, ML-Credal-NN is a lazy approach to MLC that does not use any training phase.

Let $num\_neighbors$ be the number of neighbors considered and $\mathbf{x}$ the attribute vector of an instance to classify. ML-Credal-NN, as ML-NN, computes the $num\_neighbors$-nearest neighbors of the instance by using a distance function on the attribute space. For each label y_j, with $1 \leqslant j \leqslant n_L$, let $\mathcal{K}_j(\mathbf{x})$ denote the number of neighbors of the instance (among the $num\_neighbors$-nearest ones) for which y_j is relevant. ML-Credal-NN mainly differs from ML-NN in the estimation of the prior probability that y_j is relevant (irrelevant) for the instance, namely $P^{ML-CNN}(y_j)$ ($P^{ML-CNN}(\overline{y_j})$), and the posterior probability that the instance has $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j conditioned on y_j is relevant (irrelevant) for the instance, namely $P^{ML-CNN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$ ($P^{ML-CNN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$).

With regard to the prior probabilities, ML-Credal-NN considers, for each label y_j, with $1 \leqslant j \leqslant n_L$, the A-NPI-M probability interval on the training set:

$$I^{train}_{ANPI}(y_j) = \left[\max\left(\frac{n_{tr}(y_j) - 1}{N_{tr}}, 0\right), \min\left(\frac{n_{tr}(y_j) + 1}{N_{tr}}, 1\right)\right], \tag{11.11}$$

where N_{tr} is the total number of instances in the training set and $n_{tr}(y_j)$ is the number of training instances for which y_j is relevant.

The following credal set corresponds to this interval:

$$\mathcal{P}^{train}_{ANPI}(y_j) = \left\{p \in \mathcal{P}(y_j) \mid p(y_j) \in I^{train}_{ANPI}(y_j)\right\}, \tag{11.12}$$

$\mathcal{P}(y_j)$ being the set of all probability distributions on y_j, $\quad \forall j = 1, 2, \ldots, n_L$.

Uncertainty measures can be applied to the credal set $\mathcal{P}^{train}_{ANPI}(y_j)$. As we know, the maximum entropy is a suitable uncertainty measure on credal sets since it satisfies the crucial properties. For this reason, the prior probability estimated by ML-Credal-NN that y_j is relevant for $\mathbf{x}$, $\hat{p}^{ML-CNN}(y_j)$, is the one that attains the maximum entropy on $\mathcal{P}^{train}_{ANPI}(y_j)$. Let $n_{tr}(\overline{y_j})$ denote the number of training instances for which y_j is irrelevant. If we apply Algorithm 14, we may deduce that $\hat{p}^{ML-CNN}(y_j)$ is determined as follows:

$$\hat{p}^{ML-CNN}(y_j) = \begin{cases} \frac{1}{2} & \text{if} \quad \left|n_{tr}(y_j) - n_{tr}(\overline{y_j})\right| \leqslant 2 \\ \frac{n_{tr}(y_j)-1}{N_{tr}} & \text{if} \quad n_{tr}(y_j) > n_{tr}(\overline{y_j}) + 2 \\ \frac{n_{tr}(y_j)+1}{N_{tr}} & \text{if} \quad n_{tr}(\overline{y_j}) > n_{tr}(y_j) + 2 \end{cases} \tag{11.13}$$

For each $j = 1, 2, \dots, n_L$, let $\delta_j(k_j)$ denote the number of training instances for which y_j is relevant and have k_j neighbors that have associated y_j, and $\delta'_j(k_j)$ the number of training instances that have no associated y_j and have k_j neighbors for which y_j is relevant, $\quad \forall k_j = 0, 1, \dots, num\_neighbors$. For the estimation of the posterior probabilities, ML-Credal-NN considers the following A-NPI-M credal sets on $\{0, 1, \dots, num\_neighbors\}$:

$$\begin{aligned} \mathcal{P}^{y_j}_{ANPI} = & \left\{ p \mid \sum_{k_j=0}^{num\_neighbors} p(k_j) = 1,\ l^1_{k_j} \leqslant p(k_j) \leqslant u^1_{k_j}, \right. \\ & \left. \forall k_j = 0, 1, \dots, num\_neighbors \right\}, \\ \mathcal{P}^{\overline{y_j}}_{ANPI} = & \left\{ p \mid \sum_{k_j=0}^{num\_neighbors} p(k_j) = 1,\ l^2_{k_j} \leqslant p(k_j) \leqslant u^2_{k_j}, \right. \\ & \left. \forall k_j = 0, 1, \dots, num\_neighbors \right\}, \end{aligned} \tag{11.14}$$

where $l^1_{k_j} = \max\left(\frac{\delta_j(k_j)-1}{n_{tr}(y_j)}, 0\right)$, $l^2_{kj} = \max\left(\frac{\delta'_j(k_j)-1}{n_{tr}(\overline{y_j})}, 0\right)$, $u^1_{kj} = \min\left(\frac{\delta_j(k_j)+1}{n_{tr}(y_j)}, 1\right)$, $u^2_{kj} = \min\left(\frac{\delta'_j(k_j)+1}{n_{tr}(\overline{y_j})}, 1\right)$, $\quad \forall k_j = 0, 1, \dots, num\_neighbors, \quad j = 1, 2, \dots, n_L$.

Again, ML-Credal-NN considers the probability distributions that attain the maximum entropies on $\mathcal{P}^{y_j}_{ANPI}$ and $\mathcal{P}^{\overline{y_j}}_{ANPI}$, denoted by $\hat{p}^{ML-CNN}_1$ and $\hat{p}^{ML-CNN}_2$, respectively. They can be obtained through Algorithm 14, our proposed procedure to obtain the probability distribution of maximum entropy on an A-NPI-M credal set. In this way, ML-Credal-NN estimates $P^{ML-CNN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$ and $P^{ML-CNN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$ by means of $\hat{p}^{ML-CNN}_1$ and $\hat{p}^{ML-CNN}_2$, respectively. Therefore, the set of labels predicted by ML-Credal-NN as relevant for the instance is given by:

$$\begin{aligned} h^{ML-CNN}(\mathbf{x}) = & \left\{ y_j \mid \hat{p}^{ML-CNN}(y_j) \hat{p}^{ML-CNN}_1 \left(\mathcal{K}_j(\mathbf{x})\right) > \right. \\ & \left. \left(1 - \hat{p}^{ML-CNN}(y_j)\right) \hat{p}^{ML-CNN}_2 \left(\mathcal{K}_j(\mathbf{x})\right), \quad 1 \leqslant j \leqslant n_L \right\}. \end{aligned} \tag{11.15}$$

The posterior probability predicted by ML-Credal-NN about the relevance of y_j for the instance is determined as follows:

$$f^{ML-CNN}(\mathbf{x}, y_j) = \frac{\hat{p}^{ML-CNN}(y_j)\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right)}{\hat{p}^{ML-CNN}(y_j)\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) + \left(1 - \hat{p}^{ML-CNN}(y_j)\right)\hat{p}_2^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right)}. \quad (11.16)$$

11.4.2 Binary Relevance Credal Nearest Neighbors

The Binary Relevance Credal Nearest Neighbors method, (BR-Credal-NN), proposed here, similar to BR-NN, considers a binary classification task per label.

Let $num\_neighbors$ be the number of neighbors considered. Suppose that it is wanted to classify an instance with attribute vector $\mathbf{x}$. BR-Credal-NN computes the $num\_neighbors$-nearest neighbors of the instance via a distance function on the attribute space. For each label y_j, with $1 \leqslant j \leqslant n_L$, let $\mathcal{K}_j(\mathbf{x})$ denote the number of neighbors of $\mathbf{x}$ that have associated y_j and $\overline{\mathcal{K}_j(\mathbf{x})}$ the number of neighbors of $\mathbf{x}$ for which y_j is irrelevant. BR-Credal-NN considers the following A-NPI-M probability interval on y_j:

$$I^{\mathbf{x}}_{ANPI}(y_j) = \left[\max\left(\frac{\mathcal{K}_j(\mathbf{x}) - 1}{num\_neighbors}, 0\right), \min\left(\frac{\mathcal{K}_j(\mathbf{x}) + 1}{num\_neighbors}, 1\right)\right]. \quad (11.17)$$

This interval has associated with it the following credal set on y_j:

$$\mathcal{P}^{\mathbf{x}}_{ANPI}(y_j) = \left\{p \in \mathcal{P}(y_j) \mid p(y_j) \in I^{\mathbf{x}}_{ANPI}(y_j), \quad \forall j = 1, 2, \ldots, n_L\right\}. \quad (11.18)$$

BR-Credal-NN considers the probability distribution of maximum entropy on this credal set, namely $\hat{p}^{BR-CNN}_{j,\mathbf{x}}$. Such a probability distribution can be obtained via Algorithm 14. Thus, the posterior probability predicted by BR-Credal-NN that y_j is relevant for the instance is given by:

$$f^{BR-CNN}(\mathbf{x}, y_j) = \hat{p}^{BR-CNN}_{j,\mathbf{x}}. \quad (11.19)$$

The set of labels predicted by BR-Credal-NN as relevant for the instance is composed of those labels for which the predicted posterior probability is greater or equal than 0.5:

$$h^{BR-CNN}(\mathbf{x}) = \left\{y_j \mid f^{BR-CNN}(\mathbf{x}, y_j) \geqslant 0.5, \quad 1 \leqslant j \leqslant n_L\right\}. \quad (11.20)$$

11.4.3 Justification of the proposed lazy multi-label classifiers

In this subsection, we show the principal advantage of our proposed lazy MLC algorithms, based on the A-NPI-M, versus the ones proposed so far that employ precise probabilities: the former algorithms are more suitable than the latter to tackle the class-imbalance problem that tends to arise in MLC.

ML-NN vs ML-Credal-NN: Both ML-NN and ML-Credal-NN use the MAP to predict whether a label is relevant or irrelevant for an instance, considering the number of neighbors that have associated that label. However, whereas ML-NN uses probabilities estimated by relative frequencies with Laplacian correction, ML-Credal-NN utilizes probability distributions that attain the maximum entropy on A-NPI-M credal sets.

To estimate the prior probabilities, the whole training set is employed in both algorithms. It is easy to observe that, when the sample size is very large, the probability distributions that attain the maximum entropy with the A-NPI-M do not differ much from the ones estimated with Laplacian correction. Thereby, the estimations of the prior probabilities in ML-NN and ML-Credal-NN are not very different. Nevertheless, for the estimation of the posterior probability conditioned on the label is relevant (irrelevant), only the training instances for which the label is relevant (irrelevant) are taken into account. In consequence, the performances of ML-NN and ML-Credal-NN principally differ on the estimation of the posterior probabilities, especially the probability conditioned on the label is relevant as, in MLC, there are usually very few instances that have associated a certain label. Let $P^{ML-NN}(y_j)$, $P^{ML-NN}(\overline{y_j}) = 1 - P^{ML-NN}(y_j)$, $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$, and $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$ denote the probabilities estimated by ML-NN and $\hat{p}^{ML-CNN}(y_j)$, $\hat{p}^{ML-CNN}(\overline{y_j}) = 1 - \hat{p}^{ML-CNN}(y_j)$, $\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right)$, and $\hat{p}_2^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right)$ the probabilities estimated by ML-Credal-NN.

Remark that, due to the class imbalance of MLC datasets, MLC algorithms normally predict that a label is irrelevant for an instance. For ML-NN, $P^{ML-NN}(\overline{y_j})$ is often quite higher than $P^{ML-NN}(y_j)$. Hence, a label y_j is not predicted as relevant by ML-NN unless $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$ is much greater than $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j})$. The same happens with ML-Credal-NN.

The following result shows that, if $\mathcal{K}_j(\mathbf{x})$ is large enough, then $\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right)$ is higher than $P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j)$.

Proposition 11.4.1 *If* $\left(\mathcal{K}_j(\mathbf{x}) - 1\right) \mathrm{num\_neighbors} \geqslant 2n_{tr}(y_j)$ *then*

$$\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) \geqslant p_L(\mathcal{K}_j(\mathbf{x}) \mid y_j).$$

Proof: Under our hypothesis,

$$n_{tr}(y_j)\mathcal{K}_j(\mathbf{x}) - n_{tr}(y_j) + \text{num\_neighbors}\mathcal{K}_j(\mathbf{x}) - \text{num\_neighbors} \geqslant$$
$$n_{tr}(y_j)\mathcal{K}_j(\mathbf{x}) + n_{tr}(y_j) \Rightarrow$$
$$\frac{\mathcal{K}_j(\mathbf{x}) - 1}{n_{tr}(y_j)} \geqslant \frac{\mathcal{K}_j(\mathbf{x}) + 1}{n_{tr}(y_j) + \text{num\_neighbors}}.$$

In this way,

$$\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) \geqslant$$
$$\frac{\mathcal{K}_j(\mathbf{x}) - 1}{n_{tr}(y_j)} \geqslant \frac{\mathcal{K}_j(\mathbf{x}) + 1}{n_{tr}(y_j) + \text{num\_neighbors}} = P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j).$$

□

Thus, ML-Credal-NN predicts relevant labels as relevant more frequently than ML-NN. The following example illustrates this issue:

Example 11.4.1 *Suppose that we have* $N_{tr} = 100$ *training instances and that, for a label* y_j*,* $n_{tr}(y_j) = 10$ *and* $n_{tr}(\overline{y_j}) = 90$*. Let us fix* $\text{num\_neighbors} = 10$*. Let* $\mathbf{x}$ *be the attribute vector of an instance to classify. Let us assume that there are* 6 *training instances that have associated* y_j *and have* $\mathcal{K}_j(\mathbf{x})$ *neighbors for which* y_j *is relevant and that* 4 *training instances have no associated* y_j *and have* $\mathcal{K}_j(\mathbf{x})$ *neighbors for which* y_j *is relevant. In such a case:*

$$P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) = \frac{11}{102} \times \frac{7}{21} <$$
$$\frac{91}{102} \times \frac{5}{101} = P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}),$$
$$\hat{p}^{ML-CNN}(y_j)\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) = \frac{11}{100} \times \frac{5}{10} >$$
$$\frac{89}{100} \times \frac{5}{90} \geqslant \hat{p}^{ML-CNN}(\overline{y_j})\hat{p}_2^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right).$$

Consequently, in this case, y_j *is predicted as irrelevant for* $\mathbf{x}$ *by ML-NN and as relevant by ML-Credal-NN.*

If noise is introduced in the training set by changing the value of a label for an instance from relevant to irrelevant, then ML-NN might be more affected by this fact than ML-Credal-NN, as shown in the following example:

Example 11.4.2 *Suppose that there are* $N_{tr} = 100$ *training instances. Let* y_j *be a label such that* $n_{tr}(y_j) = 10$ *and* $n_{tr}(\overline{y_j}) = 90$*. Let us fix* $\text{num\_neighbors} = 10$*. Let* $\mathbf{x}$ *denote the attribute vector of an instance that is required to be classified. Let* *us assume that there are 6 instances in the training set for which y_j is relevant and have $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j, and 2 training instances that have no associated y_j and have $\mathcal{K}_j(\mathbf{x})$ neighbors for which y_j is relevant. In this case,*

$$P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) = \frac{11}{102} \times \frac{7}{21} >$$
$$\frac{91}{102} \times \frac{3}{101} = P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}),$$
$$\hat{p}^{ML-CNN}(y_j)\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) = \frac{11}{100} \times \frac{5}{10} > \frac{89}{100} \times \frac{3}{90} \geqslant$$
$$\hat{p}^{ML-CNN}(\overline{y_j})\hat{p}_2^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right).$$

Now, suppose that noise is introduced by changing, for a training instance that has $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j, the value of y_j from relevant to irrelevant. In this noisy dataset, $n_{tr}(y_j) = 9$, $n_{tr}(\overline{y_j}) = 91$, there are 5 instances for which y_j is relevant and have $\mathcal{K}_j(\mathbf{x})$ neighbors that have associated y_j and 3 instances for which y_j is irrelevant and have $\mathcal{K}_j(\mathbf{x})$ neighbors for which y_j is relevant. We have that:

$$P^{ML-NN}(y_j)P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid y_j) = \frac{10}{102} \times \frac{6}{20} <$$
$$\frac{92}{102} \times \frac{4}{102} = P^{ML-NN}(\overline{y_j})P^{ML-NN}(\mathcal{K}_j(\mathbf{x}) \mid \overline{y_j}),$$
$$\hat{p}^{ML-CNN}(y_j)\hat{p}_1^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right) = \frac{10}{100} \times \frac{4}{9} >$$
$$\frac{90}{100} \times \frac{4}{91} \geqslant \hat{p}^{ML-CNN}(\overline{y_j})\hat{p}_2^{ML-CNN}\left(\mathcal{K}_j(\mathbf{x})\right).$$

Hence, with the original dataset, both ML-NN and ML-Credal-NN predict that $\mathbf{x}$ *has associated y_j. Nonetheless, with the noisy dataset, ML-NN changes that prediction, while ML-Credal-NN keeps predicting that y_j is relevant for* $\mathbf{x}$.

Therefore, we can deduce that ML-Credal-NN is more appropriate than ML-NN to address the class-imbalance problem that frequently appears in MLC, especially when data contains labels noise.

BR-NN vs BR-Credal-NN: To classify a new instance with attribute vector $\mathbf{x}$, BR-NN predicts that the instance has associated a label y_j if, and only if, $\mathcal{K}_j(\mathbf{x}) \geqslant \frac{num\_neighbors}{2}$, whereas BR-Credal-NN predicts that y_j is relevant for the instance if, and only if, $\mathcal{K}_j(\mathbf{x}) + 1 \geqslant \frac{num\_neighbors}{2}$. In consequence, if a label is predicted as relevant for the instance by BN-NN, then it is also predicted as relevant by BR-Credal-NN, but not vice-versa. Thus, BR-Credal-NN probably predicts more relevant labels as relevant than BR-NN. This is an important point in favor of BR-Credal-NN because, in MLC datasets, very few instances have often associated a certain label and, thus, it is little probable that the label is relevant for at least half of the neighboring instances.

When noise is introduced in the neighborhood of the instance, it is even more difficult to detect the relevant labels for BR-NN than for BR-Credal-NN. We illustrate this point in the following example:

Example 11.4.3 *Let* $\mathbf{x}$ *be the attribute vector of an instance that is required to be classified. Suppose that* 5 *neighbors of the instance have associated a label* y_j *and, for the others* 5 *neighbors,* y_j *is irrelevant. In this case, both BR-NN and BR-Credal-NN predict that the instance has associated* y_j*.*

Nevertheless, if the value of y_j *for a neighboring instance is changed from relevant to irrelevant, then BR-NN changes its prediction, whereas BR-Credal-NN keeps predicting that the instance has associated* y_j*.*

In this way, BR-Credal-NN may be more suitable than BR-NN to address the class-imbalance problem that tends to appear in MLC, especially when data contain label noise.

11.4.4 Experiments

In this experimental analysis, we aim to compare the performance of the proposed lazy MLC algorithms, based on the A-NPI-M, with the existing ones based on classical probability theory.

11.4.4.1 *Experimental settings*

- **Datasets**: Twenty datasets have been used in this experimentation, which can be downloaded from the official website of Mulan.[5] Table 11.5 shows the most important characteristics of each dataset: number of instances, number and continuous and discrete attributes, number of labels, label cardinality, and label density. We may note that the datasets are diverse concerning these issues. So, we can state that the datasets employed in this experimentation are representative.

- **Algorithms**: We have employed five algorithms in our experiments: the already existing ML-NN, BR-NN, and DML-NN, exposed in Section 6.7, and the proposed ML-Credal-NN and BR-Credal-NN. The IBLR-ML and MLCW-NN algorithms have not been considered here because of their high computational cost.

[5] http://mulan.sourceforge.net/datasets-mlc.html

Table 11.5: Datasets employed in the experimental analysis with lazy MLC methods. N is the number of instances, N_CA (N_DA) is the number of continuous (discrete) attributes, N_L is the number of labels, and L_C (L_D) is the label cardinality (density).

Dataset	Domain	N	N_CA	N_DA	N_L	L_C	L_D
bibtex	Text	7395	0	1836	159	2.4	0.015
birds	Multimedia	645	258	2	19	1.014	0.053
business	Text	11214	21924	0	30	1.599	0.053
cal500	Multimedia	502	68	0	174	26.044	0.150
computers	Text	12444	34096	0	33	1.507	0.046
corel5k	Multimedia	5000	0	499	374	3.52	0.009
education	Text	12030	57534	0	33	1.463	0.044
emotions	Multimedia	593	72	0	6	1.87	0.311
enron	Text	1702	0	1001	53	3.378	0.064
entertainment	Text	12370	32001	0	21	1.414	0.067
flags	Multimedia	194	10	9	7	3.392	0.485
genbase	Biology	662	0	1186	27	1.252	0.046
health	Text	9250	30605	0	32	1.644	0.051
mediamill	Multimedia	43907	120	0	101	4.38	0.043
medical	Text	978	0	1449	45	1.24	0.028
scene	Multimedia	2407	294	0	6	1.07	0.179
science	Text	6428	37187	0	40	1.450	0.036
social	Text	12111	52350	0	39	1.279	0.033
society	Text	14512	31802	0	27	1.670	0.062
yeast	Biology	2417	103	0	14	4.24	0.303

- **Evaluation**: In order to compare the performance of the algorithms considered here, ten evaluation metrics have been utilized: Four are based on instance classification: *Subset Accuracy*, *Hamming Loss*, *Accuracy*, and *F1*; we have used two label-based classification metrics: *Micro F1* and *Macro F1*; four instance-based ranking metrics have been used: *Ranking Loss*, *Coverage*, *Average Precision* and *One Error*. These metrics were exhaustively explained in Section 6.3.

  Moreover, for each evaluation metric M_e, we have considered the corresponding Equalized Loss EL_{M_e}. It is based on the ELA measure for standard classification, described in Section 4.2.1.2. EL_{M_e} indicates the sensitivity to the label noise of an algorithm for the M_e measure.

  For metrics such that a lower value implies a better performance, EL_{M_e} is obtained as follows:

$$EL_{M_e} = \frac{M_e^{10}}{M_e^0},$$

  M_e^0 being the value of M_e obtained without noise in the labels and M_e^{10} the value obtained with a 10% of label noise.

For evaluation metrics such that the performance is better as its value is higher, $EL_{M_e} = \frac{1-M_e^{10}}{M_e^0}$.

The algorithms have been also compared in terms of computational time.

- **Procedure:** Two label noise levels have been considered in this experimental study: 0% and 10%. A cross-validation procedure of five folds has been carried out for each dataset and noise level. An iteration has been done for each fold, in which the corresponding partition has been used for testing and the rest of the data for training. The noise has been added to the training set as follows: for each label, the x% of the instances in the training set have been chosen and the value of the corresponding label has been changed, (if the label is relevant for the instance it has been changed to irrelevant and vice-versa), x being the noise level. The same partitions have been used for all algorithms in all datasets.

- **Software and parameters:** We have started from the implementations provided in Mulan for the ML-NN, BR-NN and DML-NN algorithms. The necessary methods for using ML-Credal-NN and BR-Credal-NN have been implemented. For all algorithms, we have used the value 10 for the parameter $num\_neighbors$ since it is the one given by default in Mulan and one of the most utilized in the literature. For both BR-NN and BR-Credal-NN, the α extension has been considered.[6]

- **Statistical evaluation**: For each evaluation measure, we compare the results obtained by the algorithms without noise in the labels and in the corresponding Equalized Loss. For this purpose, following the recommendations given in [75] about statistical comparisons between the results obtained by three or more algorithms on many datasets, we use the Friedman test. When the null hypothesis of the Friedman test is rejected, we employ the Nemenyi test. The level of significance considered is $\alpha = 0.05$. We present the results of the Friedman and Nemenyi tests through critical diagrams.

11.4.4.2 *Results and discussion*

Table 11.6 shows the average Friedman rank obtained by each algorithm in each evaluation metric and computational time without noise in the labels. For each evaluation metric, the best result is marked in bold. The critical diagrams corresponding to the Nemenyi tests without noise in the labels can be seen in

[6] Both BR-NN extensions, BR-NN-α and BR-NN-β, perform equivalently [194].

Figure 11.1. The average Friedman ranks obtained by each algorithm in the metrics associated with Equalized Losses are presented in Table 11.7. In such a table, the best results are also marked in bold. Figure 11.2 lets us see the critical diagrams corresponding to the Nemenyi test associated with Equalized Losses. The critical diagram associated with computational time is presented in Figure 11.3.

Table 11.6: Average Friedman ranks obtained by the lazy MLC methods without noise in the labels for each metric.

Metric	ML-NN	BR-NN	DML-NN	ML-Credal-NN	BR-Credal-NN
Hamming Loss	2	4.05	2.7	**1.55**	4.7
Subset Accuracy	3.6	**1.95**	3.95	2.6	2.9
Accuracy	3.65	2.95	4.2	2.4	**1.8**
F1	3.65	3.1	4.2	2.4	**1.65**
Micro F1	3.45	3.45	4.1	2.55	**1.45**
Macro F1	3.45	3.15	4.5	2.35	**1.55**
Coverage	1.75	3.975	**1.65**	2.95	4.675
Ranking Loss	1.8	3.975	**1.55**	2.95	4.725
Average Precision	**1.8**	4	2.15	2.3	4.875
One Error	**2**	3.825	2.4	2.25	4.525
Computational Time	3.55	**1.5**	4.9	3.55	**1.5**

Table 11.7: Average Friedman ranks for Equalized Losses corresponding to each metric obtained by the lazy MLC methods.

Metric	ML-NN-NN	BR-NN	DML-NN	ML-Credal-NN	BR-Credal-NN
Hamming Loss	2.4	3.05	**2.3**	2.6	4.65
Subset Accuracy	3.6944	**1.7222**	4.1389	2.4444	3
Accuracy	3.65	2.7	4.25	2.6	**1.8**
F1	3.6	2.9	4.25	2.7	**1.55**
Micro F1	3.6	2.7	4.35	2.4	**1.95**
Macro F1	3.6	2.95	4.4	2.3	**1.75**
Coverage	2.85	**2.65**	2.75	3.25	3.5
Ranking Loss	**2.6**	2.8	**2.6**	3.35	3.65
Average Precision	**1.85**	3.5	2.7	2.4	4.55
One Error	3.05	**2.475**	3.15	3	3.325

About the obtained results, we remark the following points for each type of evaluation metric:

Classification-based metrics:

- Regarding Hamming Loss, which measures the differences between the real and predicted sets of labels, the lowest Friedman ranks are achieved by ML-Credal-NN and ML-NN. Both BR-NN and BR-Credal-NN obtain very poor performance in this evaluation measure since they obtain the

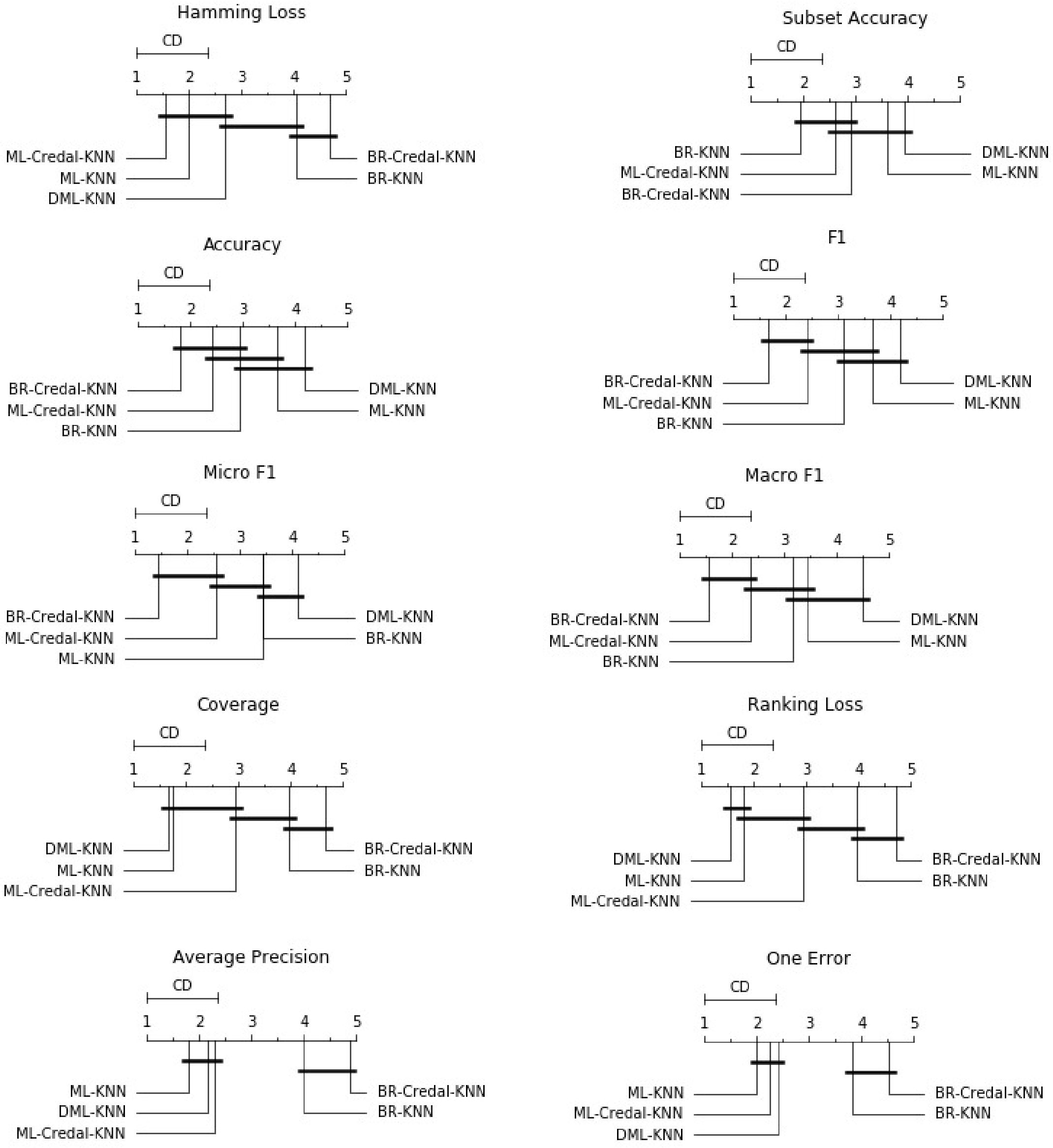

Figure 11.1: Critical diagrams corresponding to the Friedman and Nemenyi tests without noise in the labels in the experimental study with lazy MLC algorithms. CD = Critical Distance.

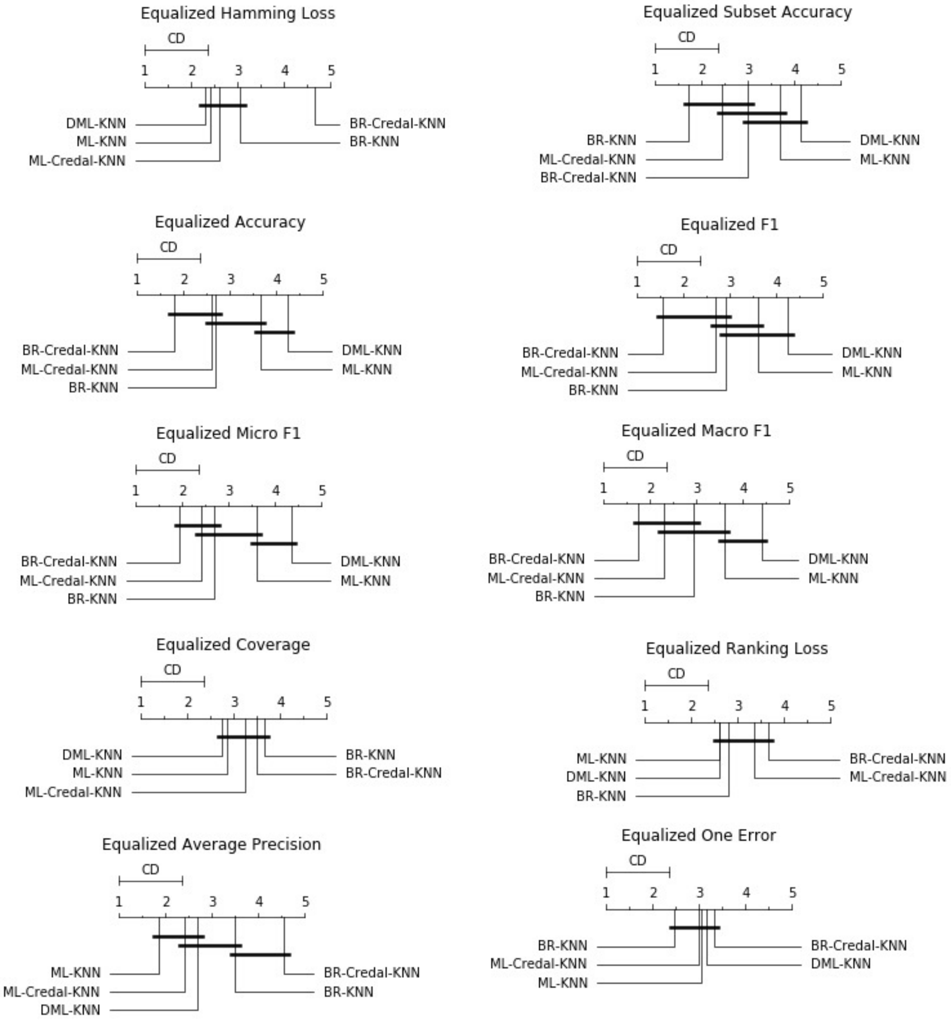

Figure 11.2: Critical diagrams associated with Equalized Loss for each evaluation metric in the experimental analysis with lazy MLC methods. CD = Critical Distance

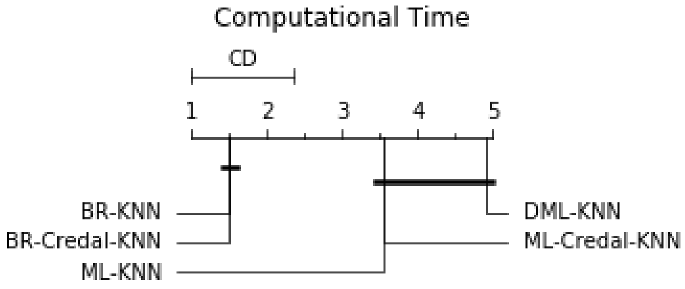

Figure 11.3: Critical diagram corresponding to computational time in the experimental study with lazy MLC algorithms. CD = Critical Distance

highest Friedman ranks and are not connected via segments with ML-NN and ML-Credal-NN in the critical diagram associated with Hamming Loss, which implies that BR-NN and BR-Credal-NN are significantly outperformed by ML-NN and ML-Credal-NN in Hamming Loss according to the Nemenyi test. In addition, in the critical diagram corresponding to Hamming Loss, DML-NN and BR-Credal-NN are not connected by means of a segment. Thus, DML-NN performs significantly better than BR-Credal-NN in Hamming Loss according to the Nemenyi test. Due to the results obtained in Equalized Loss Hamming Loss, we can state that BR-Credal-NN is, by far, the most sensitive to the noise in this metric. Indeed, this method obtains the highest average Friedman rank in Equalized Loss Hamming Loss, and it is not connected via segments with the other algorithms in the critical diagram associated with Equalized Loss Hamming Loss. Consequently, BR-Credal-NN is significantly outperformed by the other algorithms via the Nemenyi test in Equalized Loss Hamming Loss. The rest of the lazy MLC algorithms considered here perform equivalently according to the Nemenyi test in terms of robustness to noise for Hamming Loss as they are connected with a segment in the corresponding critical diagram.

- The BR-NN algorithm achieves the highest proportion of instances for which the predicted set of labels coincides with the set of relevant labels as it obtains the lowest Friedman rank in Subset Accuracy. Furthermore, in the critical diagram corresponding to this metric, BR-NN is not connected via segments with DML-NN and ML-NN. Hence, BR-NN performs significantly better than DML-NN and ML-NN according to the Nemenyi test in Subset Accuracy. These are the only pairs of algorithms that are not connected with segments in the critical diagram of

Subset Accuracy. Thereby, there are no more cases of statistically significant differences between pairs of algorithms in this metric, although our proposed lazy MLC algorithms obtain lower Friedman ranks than ML-NN and DML-NN. According to the Equalized Loss Subset Accuracy results, BR-NN is the most robust to noise in this metric, followed by ML-Credal-NN. Indeed, BR-NN and ML-Credal-NN obtain the lowest Friedman ranks in Equalized Loss Subset Accuracy. Moreover, in the critical diagram associated with Equalized Loss Subset Accuracy, these algorithms are not connected through a segment with DML-NN. This means that BR-NN and ML-Credal-NN significantly outperform DML-NN according to the Nemenyi test in Equalized Loss Subset Accuracy. In the critical diagram corresponding to Equalized Loss Subset Accuracy, BR-NN and ML-NN are not connected via a segment and, thus, the former algorithms performs significantly better than the latter in Equalized Loss Subset Accuracy according to the Nemenyi test.

- In Accuracy, the best Friedman rank is achieved by BR-Credal-NN, followed by ML-Credal-NN. In the critical diagram corresponding to this metric, DML-NN is not connected with BR-Credal-NN nor with ML-Credal-NN through segments. The same happens with ML-NN and BR-Credal-NN. Therefore, BR-Credal-NN and ML-Credal-NN perform significantly better than DML-NN according to the Nemenyi test in Accuracy, and BR-Credal-NN also outperforms ML-NN via the Nemenyi test in this metric. With regard to the sensitivity to noise for this metric, the best Friedman ranks are obtained by BR-Credal-NN and the worst by ML-NN and DML-NN. Furthermore, in the critical diagram corresponding to Equalized Loss Accuracy, BR-Credal-NN is not connected via segments with ML-NN nor with DML-NN. So, BR-Credal-KNN performs significantly better than DML-NN and ML-NN in Equalized Loss Accuracy according to the Nemenyi test. Also, in terms of robustness to noise for Accuracy, DML-NN is significantly outperformed by ML-Credal-NN and BR-NN as DML is not connected via segments with ML-Credal-NN nor with BR-NN in the critical diagram corresponding to Equalized Loss Accuracy.

- Concerning the F1 metrics, the lowest Friedman ranks are obtained, again, by BR-Credal-NN, followed by ML-Credal-NN. In the critical diagrams associated with the F1 evaluation measures, BR-Credal-NN is connected via a segment only with ML-Credal-NN. Consequently, BR-Credal-NN significantly outperforms all the algorithms considered here via the Nemenyi test in the F1 metrics, except for ML-Credal-NN. In addition, ML-Credal-NN performs significantly better than DML-NN in the F1 metrics as these two algorithms are not connected via segments in the corresponding critical diagrams. The results obtained in Equalized Losses associated with the F1 measures allow deducing that BR-Credal-NN is the most robust to noise in these metrics. Indeed, it obtains the lowest Friedman ranks in the Equalized Losses associated with the F1 metrics and, in the corresponding critical diagrams, it is not connected through segments with DML-NN nor with ML-NN. Therefore, BR-Credal-NN obtains significantly better results than DML-NN and ML-NN according to the Nemenyi test in Equalized Losses associated with the F1 metrics. Also, ML-NN is significantly more sensitive to noise than ML-Credal-NN and BR-NN in the F1 measures because the Friedman ranks obtained by ML-NN in the Equalized Losses corresponding to the F1 metrics are higher than the ones obtained by ML-Credal-NN and BR-NN and the former algorithm is not connected via segments with the other two algorithms in the corresponding critical diagrams.

Ranking-based metrics:

- BR-Credal-NN obtains the worst Friedman ranks in all the ranking-based measures considered here, followed by BR-NN. None of these two algorithms is connected with DML-NN via a segment in none of the critical diagrams corresponding to the ranking-based evaluation measures. Thus, both BR-Credal-NN and BR-NN are significantly outperformed by DML-NN according to the Nemenyi test in all ranking-based metrics. ML-Credal-NN and ML-NN also perform significantly better than BR-Credal-NN in these measures via the Nemenyi test since the former algorithms are not connected via segments with the latter in the critical diagrams corresponding to the ranking-based metrics.
- In Coverage, which measures the average number of steps that are necessary to go down the label ranking to cover all relevant labels for an instance, the lowest Friedman ranks are obtained by DML-NN and ML-NN. In the critical diagram associated with this metric, none of these two algorithms is connected through segments with BR-NN nor with BR-Credal-NN. Hence, DML-NN and ML-NN perform significantly better than BR-NN and BR-Credal-NN according to the Nemenyi test in Coverage. The only algorithm that is not connected with ML-Credal-NN through a segment in the critical diagram associated with Coverage is BR-Credal-NN. In consequence, according to the Nemenyi test,

ML-Credal-NN only significantly outperforms BR-Credal-NN in Coverage. The results are similar in Ranking Loss, which indicates the average proportion of pairs of relevant-irrelevant labels reversely ordered in the label ranking, but now DML-NN performs significantly better than ML-Credal-NN as these two algorithms are not connected with a segment in the critical diagram corresponding to Ranking Loss.

- Regarding Average Precision and One error, which indicate, respectively, the average number of labels ranked above a relevant one and the proportion of instances for which the top-ranked label is not relevant, ML-NN, DML-NN, and ML-Credal-NN, obtain statistically equivalent results. In fact, these three algorithms are connected via segments in the critical diagrams associated with these metrics. Also, in such critical diagrams, these three algorithms are not connected through segments with BR-NN nor with BR-Credal-NN. Thereby, ML-NN, DML-NN, and ML-Credal-NN significantly outperform BR-NN and BR-Credal-NN via the Nemenyi test in Average Precision and One Error.

- The results do not vary much with a 10% of noise in the labels; the results obtained by the five algorithms considered in this experimentation in Equalized Losses associated with Coverage, Ranking Loss, and One Error, are statistically equivalent according to the Nemenyi test. Actually, the five algorithms are connected through segments in the corresponding critical diagrams. In Equalized Loss Average Precision, ML-NN performs significantly better than BR-Credal-NN and BR-NN according to the Nemenyi test as the former algorithm achieves a lower Friedman rank in this metric and is not connected with the other two algorithms via segments in the associated critical diagram. DML-NN and ML-Credal-NN also significantly outperform BR-Credal-NN in terms of sensitivity to noise in Average Precision. Indeed, in Equalized Loss Average Precision, DML-NN and ML-Credal-NN obtain lower Friedman ranks than BR-Credal-NN, and the latter method is not connected with the other two algorithms via segments in the critical diagram corresponding to Equalized Loss Average Precision.

Computational time:

- DML-NN is, by far, the algorithm that requires the highest computational cost. Indeed, this method obtains the highest Friedman rank in computational time. Moreover, DML-NN is not connected with the other algorithms via segments in the critical diagram associated with computational time. In consequence, there are statistically significant differences according to the Nemenyi test between the computational time required by DML-NN and the computational times required by the other algorithms.

- The computational times of ML-NN and ML-Credal-NN are statistically equivalent. In fact, these two algorithms are connected with a segment in the critical diagram associated with computational time. The Friedman ranks obtained by these algorithms in computational time are lower than the one achieved by DML-NN, even though there are no statistically significant differences via the Nemenyi test (both ML-NN and ML-Credal-NN are connected through a segment with DML-NN in the critical diagram associated with computational time).

- BR-NN and BR-Credal-NN obtain lower Friedman ranks than ML-NN and ML-Credal-NN in computational time. In addition, in the corresponding critical diagram, BR-NN and BR-Credal-NN are not connected via segments with ML-NN and ML-Credal-NN. Hence, in terms of computational time, ML-NN and ML-Credal-NN are significantly outperformed by the lazy BR-based methods according to the Nemenyi test.

Summary of the results: The results obtained in this experimental analysis can be summarized in the following issues:

- For all F1 measures, the ML-Credal-NN and BR-Credal-NN algorithms, proposed in this section, achieve the best performance. Furthermore, they are the more robust to noise in F1 metrics. The same happens with Accuracy. It is since, as argued in Section 11.4.3, our proposed lazy MLC algorithms, based on the A-NPI-M, predict more relevant labels as relevant than the existing ones based on classical probability theory, especially when data contain label noise.

- BR-Credal-NN performs slightly better than ML-Credal-KNN in Accuracy and F1-based measures. However, the performance of BR-Credal-NN in Hamming Loss is quite poor. In classification-based metrics, DML-NN generally obtains the worst results.

- With regard to ranking-based measures, both BR-NN and BR-Credal-NN perform worse than the other algorithms considered here. We can state that, in these metrics, the ML-NN, DML-NN, and ML-Credal-NN algorithms have equivalent performance.

- Concerning the computational time, the BR-based methods are the ones that achieve the best results, followed by ML-NN and ML-Credal-NN, and the DML-NN algorithm is the one that requires the highest computational time. It makes sense since the BR-based methods are quite simple, and DML-NN employs a more sophisticated MAP principle than ML-NN. The computational times of ML-NN and ML-Credal-NN, and BR-NN and BR-Credal-NN are equivalent. It is because the differences between the complexities of the estimations of the probabilities in ML-NN and ML-Credal-NN are not significant. The same occurs with BR-NN and BR-Credal-NN.

11.5 New label ordering method for Classifier Chains

Before exposing our proposed procedure to insert the labels in the chain, i.e, determine the permutation $\sigma : \{1, \dots, n_L\} \to \{1, \dots, n_L\}$ that generates the label order, we explain how the correlation between each pair of labels is estimated in our proposed label ordering method.

Let N_{tr} be the total number of instances in the training set. For each label y_j, with $1 \leqslant j \leqslant n_L$, let $n_{tr}(y_j)$ ($n_{tr}(\overline{y_j})$) denote the number of training instances for which y_j is relevant (irrelevant). We consider the following A-NPI-M probability interval on y_j corresponding to the training set:

$$I^{train}_{ANPI}(y_j) \in \left[\max\left(\frac{n_{tr}(y_j)-1}{N_{tr}}, 0\right), \min\left(\frac{n_{tr}(y_j)+1}{N_{tr}}, 1\right)\right]. \tag{11.21}$$

This interval has associated with it the following credal set:

$$\mathcal{P}^{train}_{ANPI}(y_j) = \left\{p \in \mathcal{P}(y_j) \mid p(y_j) \in I^{train}_{ANPI}(y_j)\right\}, \tag{11.22}$$

$\mathcal{P}(y_j)$ being the set of all probability distributions on y_j, $\quad \forall j = 1, 2, \dots, n_L$.

Uncertainty measures can be applied to this credal set. As we know, the maximum entropy is an appropriate uncertainty measure on this type of set since it satisfies all essential properties. Hence, our proposed label ordering method utilizes the maximum entropy on $\mathcal{P}^{train}_{ANPI}(y_j)$:

$$S^*\left(\mathcal{P}^{train}_{ANPI}(y_j)\right) = \max_{p \in \mathcal{P}^{train}_{ANPI}(y_j)} S(p), \tag{11.23}$$

$S(p)$ being the Shannon entropy on the probability distribution p.

Applying Algorithm 14, we may deduce that the probability distribution that attains the maximum entropy on $\mathcal{P}^{train}_{ANPI}(y_j)$, namely $\hat{p}^{tr}_{ANPI}(y_j)$, is determined by:

$$\hat{p}^{tr}_{ANPI}(y_j) = \begin{cases} \frac{1}{2} & \text{if} \quad \left|n_{tr}(y_j) - n_{tr}(\overline{y_j})\right| \leqslant 2 \\ \frac{n_{tr}(y_j)-1}{N_{tr}} & \text{if} \quad n_{tr}(y_j) > n_{tr}(\overline{y_j}) + 2 \\ \frac{n_{tr}(y_j)+1}{N_{tr}} & \text{if} \quad n_{tr}(\overline{y_j}) > n_{tr}(y_j) + 2 \end{cases} \tag{11.24}$$

For each pair of labels y_j, y_k, let $n_{tr}(y_j, y_k)$ denote the number of training instances for which both y_j and y_k are relevant, $n_{tr}(\overline{y_j}, y_k)$ the number of training instances that have associated y_k but not y_j, $n_{tr}(y_j, \overline{y_k})$ the number of training instances for which y_j is relevant but y_k irrelevant, and $n_{tr}(\overline{y_j}, \overline{y_k})$ the number of training instances for which both y_j and y_k are irrelevant.

We consider the A-NPI-M credal set corresponding to the training set on y_j conditioned on y_k is relevant:

$$\begin{aligned} \mathcal{P}^{train}_{ANPI}\left(y_j \mid y_k\right) = \Bigg\{ p \in \mathcal{P}\left(y_j \mid y_k\right) \mid \max\left(0, \frac{n_{tr}(y_j, y_k) - 1}{n_{tr}(y_k)}\right) \leqslant \\ p(y_j \mid y_k) \leqslant \min\left(\frac{n_{tr}(y_j, y_k) + 1}{n_{tr}(y_k)}, 1\right)\Bigg\}, \end{aligned} \tag{11.25}$$

where $\mathcal{P}\left(y_j \mid y_k\right)$ denotes the set of all probability distributions on y_j conditioned on y_k is relevant.

Our proposed label ordering method considers the maximum entropy on this credal set, which can also be easily determined via Algorithm 14:

$$S^*\left(\mathcal{P}^{train}_{ANPI}\left(y_j \mid y_k\right)\right) = S(\hat{p}^{tr}_{ANPI}\left(y_j \mid y_k\right)), \tag{11.26}$$

where

$$\hat{p}^{tr}_{ANPI}(y_j \mid y_k) = \begin{cases} \frac{1}{2} & \text{if} \quad \left|n_{tr}(y_j, y_k) - n_{tr}(\overline{y_j}, y_k)\right| \leqslant 2 \\ \frac{n_{tr}(y_j, y_k)-1}{n_{tr}(y_k)} & \text{if} \quad n_{tr}(y_j, y_k) > n_{tr}(\overline{y_j}, y_k) + 2 \\ \frac{n_{tr}(y_j, y_k)+1}{n_{tr}(y_k)} & \text{if} \quad n_{tr}(y_j, y_k) > n_{tr}(\overline{y_j}, y_k) + 2 \end{cases} \tag{11.27}$$

Likewise, in our proposal, the maximum entropy on the A-NPI-M credal set on y_j conditioned on y_k is irrelevant, $\mathcal{P}^{train}_{ANPI}\left(y_j \mid \overline{y_k}\right)$, is considered:

$$S^*\left(\mathcal{P}^{train}_{ANPI}\left(y_j \mid \overline{y_k}\right)\right) = S(\hat{p}^{tr}_{ANPI}\left(y_j \mid \overline{y_k}\right)), \tag{11.28}$$

where:

$$\mathcal{P}^{train}_{ANPI}\left(y_j \mid \overline{y_k}\right) = \left\{p \in \mathcal{P}\left(y_j \mid \overline{y_k}\right) \mid \max\left(0, \frac{n_{tr}(y_j, \overline{y_k}) - 1}{n_{tr}(\overline{y_k})}\right) \leqslant \right. \\ \left. p(y_j \mid \overline{y_k}) \leqslant \min\left(\frac{n_{tr}(y_j, \overline{y_k}) + 1}{n_{tr}(\overline{y_k})}, 1\right)\right\}, \tag{11.29}$$

$\mathcal{P}\left(y_j \mid \overline{y_k}\right)$ being the set of all probability distributions on y_j conditioned on y_k is irrelevant,

$$\hat{p}^{tr}_{ANPI}\left(y_j \mid \overline{y_k}\right) = \begin{cases} \frac{1}{2} & \text{if} \quad \left|n_{tr}(y_j, \overline{y_k}) - n_{tr}(\overline{y_j}, \overline{y_k})\right| \leqslant 2 \\ \frac{n_{tr}(y_j, \overline{y_k}) - 1}{n_{tr}(\overline{y_k})} & \text{if} \quad n_{tr}(y_j, \overline{y_k}) > n_{tr}(\overline{y_j}, \overline{y_k}) + 2 \\ \frac{n_{tr}(y_j, \overline{y_k}) + 1}{n_{tr}(\overline{y_k})} & \text{if} \quad n_{tr}(\overline{y_j}, \overline{y_k}) > n_{tr}(y_j, \overline{y_k}) + 2 \end{cases} \tag{11.30}$$

The method used in our proposal to estimate the correlation between two labels y_j, y_k, is based on the Imprecise Information Gain [13]:

$$\mathrm{IIG}(y_j, y_k) = S^*\left(\mathcal{P}^{train}_{ANPI}(y_k)\right) - \hat{p}^{tr}_{ANPI}(y_j) S^*\left(\mathcal{P}^{train}_{ANPI}\left(y_j \mid y_k\right)\right) - \\ \left(1 - \hat{p}^{tr}_{ANPI}(y_k)\right) S^*\left(\mathcal{P}^{train}_{ANPI}\left(y_j \mid \overline{y_k}\right)\right). \tag{11.31}$$

The estimated correlation between y_j and y_k consists of the Imprecise Symmetrical Uncertainty (ISU), which uses IIG in a normalized way:

$$\mathrm{ISU}(y_j, y_k) = \frac{2 \times \mathrm{IIG}(y_j, y_k)}{S^*\left(\mathcal{P}^{train}_{ANPI}(y_j)\right) + S^*\left(\mathcal{P}^{train}_{ANPI}(y_k)\right)}. \tag{11.32}$$

The ISU measure takes as a reference the Symmetrical Uncertainty (SU) [106], widely used for estimating the correlation between two variables in classical information theory.

In this way, our proposed method to estimate the correlation between two labels consists of the reduction of uncertainty, estimated via the A-NPI-M, of a label when the value of the other label is known. Such a reduction is measured in a normalized way by considering the initial uncertainty of both labels.

For determining the label order given by the permutation σ, i.e
$y_{\sigma(1)} > y_{\sigma(2)} > \ldots > y_{\sigma(n_L)}$, we take the following points into account:

- If the label positioned at the beginning of the chain is little correlated with the other ones, then the following classifiers may utilize irrelevant information about it. In contrast, when the label placed in the first position is highly correlated with the remaining ones, the following classifiers might use important and relevant information and, consequently, their predictive performance may improve. Thus, in our proposal, the label with the maximum average correlation with the other ones is positioned at the beginning of the chain, i.e,

$$\sigma(1) = \arg \max_{j=1,2,\dots,n_L} \sum_{k=1,k\neq j}^{n_L} ISU(y_j, y_k). \qquad (11.33)$$

- To decide the label placed in the second position, we consider, for each candidate, two issues. The first one is how correlated is the candidate label with the one already inserted; if the label inserted at the second position is not correlated with the first one, then the information provided by the first label may be irrelevant, while if the first label and the second one are highly correlated, then the information provided by the first label may be very relevant for the second one and, thus, the performance of the second classifier might improve. Secondly, similar to the first step, for each candidate, it is important that the remaining candidate labels are correlated with it for the information provided by the label to be useful for the following classifiers. For these reasons, for each candidate label to be positioned at the second place, we consider a score that consists of the sum of the correlation between the candidate label and the first one and the average correlation between the candidate label and the remaining ones not inserted yet:

$$\begin{gathered} Score_2(y_k) = ISU(y_k, y_{\sigma(1)}) + \frac{\sum_{j=1, j\neq k,\sigma(1)}^{n_L} ISU(y_j, y_k)}{n_L - 2}, \\ \forall k \in \{1,2,\dots,n_L\} \setminus \{\sigma(1)\}. \end{gathered} \qquad (11.34)$$

The chosen label is the one with the highest score.

- For inserting the labels at the remaining positions, we apply the same reasoning: In the ith position, with $2 < i \leqslant n_L$, for each candidate label, the corresponding score is computed via the sum of the average correlation among the candidate label and the ones already inserted in the chain and the average correlation between the candidate label and the remaining ones not positioned yet:

$$\begin{gathered} Score_i(y_k) = \frac{\sum_{j=1}^{i-1} ISU(y_{\sigma(j)}, y_k)}{i-1} + \\ \frac{\sum_{j\neq\sigma(1),\dots,\sigma(i-1),k} ISU(y_j, y_k)}{n_L - i}, \quad \forall k \neq \sigma(1), \dots, \sigma(i-1) \end{gathered} \qquad (11.35)$$

The label placed in the ith position is the one with the highest score according to Equation (11.35).

Therefore, once the correlation between each pair of labels is computed, our proposed greedy procedure to insert the labels in the chain is determined via Algorithm 24, where $y_\sigma = \left(y_{\sigma(1)}, y_{\sigma(2)}, \ldots, y_{\sigma(n_L)}\right)$.

Algorithm 24: Our proposed label ordering procedure.

```
Procedure **Determine label order**(labels $y_1, y_2, \ldots, y_{n_L}$, correlation
between each pairs of labels $ISU(y_j, y_k)$, with $j, k = 1, 2, \ldots, n_L$)
for $k = 1$ to $n_L$ do
    1. $Score_1(y_k) = \sum_{j=1, j \neq k}^{n_L} ISU(y_j, y_k)$.
$\sigma(1) = \arg\max_{k=1,2,\ldots,n_L} Score_1(y_k)$.
for $i = 2$ to $n_L$ do
    for $k \neq \sigma(1), \ldots, \sigma(i-1)$ do
        $Score_i(y_k) = \frac{\sum_{j=1}^{i-1} ISU(y_{\sigma(j)}, y_k)}{i-1} + \frac{\sum_{j \neq \sigma(1), \ldots, \sigma(i-1), k} ISU(y_j, y_k)}{n_L - i}$.
    $\sigma(i) = \arg\max_{k \neq \sigma(1), \ldots, \sigma(i-1)} Score_i(y_k)$.
return $y_\sigma$.
```

11.5.1 Justification of our proposed label ordering procedure

The advantages of our proposed method to determine the label order in CC over the ones based on label correlations proposed so far can be summarized in the following way:

- The label ordering method based on ReliefF developed in [214] uses a list of labels correlated with each one. For this purpose, a correlation score between each pair of labels is computed, and a threshold is used to decide which labels are correlated with each one. It must be remarked that the choice of the threshold is a difficult question. In the experimental study carried out by the developers of the method, it was supposed that half of the total number of labels are correlated with each one, but this assumption is obviously unrealistic. Nevertheless, in our proposed method, we do not employ any threshold to decide whether a label is correlated with another one but we use correlation scores between pairs of labels. Furthermore, in the label ordering procedure based on ReliefF, to insert a label at the ith position of the chain, the one with the highest number of non-inserted labels correlated with it is selected whenever it is correlated with at least one label inserted, even though it is only correlated with one of the many labels already inserted. Hence, in that label ordering method, the correlation of a label with the ones already inserted has little influence, which is a drawback. In contrast, our proposed method equally considers the average correlation of a label with the remaining candidates and the average correlation of the candidate label with the ones already inserted in the chain.

- The greedy label ordering procedures proposed in [124] only take into account, for each candidate label to insert in the chain, its average conditional entropy-based correlation with the labels not inserted yet. As explained before, it is a shortcoming because if a label is not correlated with the previous ones according to the order established by the chain, then the corresponding classifier might use irrelevant information and, consequently, its performance may worsen.

- Indeed, in our proposed label ordering procedure, it is possible that, at a certain step, a label is chosen because it is strongly correlated with the ones already inserted though its correlation with the remaining candidate labels is not very high. However, such a correlation cannot be very low since, in such a case, that label would not be selected via our criterion. In these situations, the information provided by the chosen label for the remaining classifiers may not be very useful (although it would not imply noise because the correlation would not be close to 0). Nevertheless, the classifier corresponding to the selected label may use very relevant information and, thus, its performance probably improves.

  Likewise, at a certain step, a label strongly correlated with the remaining candidate labels may be selected although its correlation with the labels already inserted is not very high. In these cases, the classifier corresponding to that label may not use very relevant information (although the information must not be very irrelevant because the correlation with the labels already inserted cannot be close to 0). Nonetheless, the information provided by that label for the remaining classifiers may be very useful and, thus, their performance might improve.

- Moreover, for the estimation of the correlations between pairs of labels, our proposed method utilizes the A-NPI-M, unlike the label ordering methods based on label correlations developed so far, which employ classical probability theory. The Multi-Label classifiers based on imprecise probabilities that we have proposed in the previous sections have achieved better performance than the ones that use precise probabilities, as highlighted via experimental analyses. In general, imprecise probability models are more suitable for classification than classical probability theory when there is class noise in the data. As argued in Section 11.2.3, the intrinsic noise in MLC may be higher than the intrinsic class noise in traditional classification.

In consequence, it is expected that our proposal outperforms the label ordering procedures based on label correlations proposed so far in CC. This fact is corroborated with an experimental analysis in Section 11.5.2.

11.5.2 Experimental study

11.5.2.1 *Description of the experiments*

- **Datasets**: Ten datasets have been employed in this experimentation. They can be downloaded from the official website of Mulan [202][7], except for the *Slashdot* dataset, which can be found on the website of Meka [179][8], another Java library for MLC. Table 11.8 shows the main characteristics of each dataset: number of instances, number of attributes, number of labels, label cardinality, label density, and MLC domain.

Table 11.8: Datasets used in our experimental analysis with label ordering methods in CC. N is the number of instances, N_A is the number of attributes, n_L is the number of labels, and L_C and L_D are, respectively, the label cardinality and the label density.

Dataset	N	N_A	n_L	L_C	L_D	Domain
birds	645	260	19	1.014	0.053	Audio
cal500	502	68	174	26.044	0.15	Music
emotions	593	72	6	1.87	0.311	Music
enron	1702	1001	53	3.38	0.064	Text
flags	194	9	17	3.392	0.485	Image
genbase	662	1186	27	1.252	0.046	Biology
medical	978	1449	45	1.24	0.028	Text
scene	2407	294	6	1.07	0.179	Image
slashdot	3782	1079	22	0.9096	0.413	Text
yeast	2417	103	14	4.24	0.303	Biology

As can be seen, the datasets utilized in our experimental study are varied in terms of domain, number of instances, number of features, number of labels, label density, etc.

[7] http://mulan.sourceforge.net/datasets-mlc.html

[8] https://waikato.github.io/meka/datasets

- **Evaluation metrics**: We have employed ten evaluation metrics in our experiments. Four of them are based on instance classification: Hamming Loss, Subset Accuracy, Accuracy, and F1; two label-based classification metrics have been used: Micro F1 and Macro F1; the other four evaluation measures considered here, One Error, Coverage, Ranking Loss, and Average Precision, are based on ranking. All these metrics were described in detail in Section 6.3.

- **Algorithms**: Five algorithms have been considered in our experimental analysis: Binary Relevance (BR), the original Classifier Chain method (CC), CC with the greedy procedure based on conditional entropies that obtained the best experimental results among the ones used in [124] (CondEnt_CC), CC with the label ordering method based on ReliefF proposed in [214] (ReliefF_CC), and CC with our proposed label ordering method (ImpCorr_CC).

- **Procedure**: For each algorithm and dataset, a cross-validation procedure of 5 folds has been repeated: the dataset is divided into 5 partitions and, for each one of them, an iteration is done. In it, the corresponding partition is used for testing and the rest of the dataset for training. The model is learned using the training set, and each one of the evaluation metrics is extracted with the test set. The same partitions have been employed for all algorithms in all datasets.

- **Software** and **parameters**: We have started from the implementation available in Mulan for BR and CC, and we have added the necessary structures and methods for employing CondEnt_CC, ReliefF_CC, and ImpCorr_CC. For these algorithms, the parameters given by default in Mulan have been used. As the base classifiers, we have used the Support Vector Machines based on the SMO algorithm [173], available in Weka, with default parameters.

  We have utilized part of the functionality available in Mulan to create the partitions of cross-validation.

- **Statistical evaluation**: For each evaluation metric, there are five algorithms to compare. Thus, in accordance with the recommendations given by Demšar [75] for statistical comparisons between the results obtained by more than two algorithms, the Friedman test has been employed. When the null hypothesis of the Friedman test is rejected, we use the Nemenyi test [167] to detect the cases of statistically significant differences between pairs of algorithms.

Furthermore, we have used the Friedman test for comparing the average Friedman ranks obtained by the algorithms in the evaluation metrics considered here. Again, if the null hypothesis of this test is rejected, then the Nemenyi test is employed.

The level of significance utilized in all statistical tests is $\alpha = 0.05$.

11.5.2.2 *Results and discussion*

Table 11.9 shows the average Friedman rank obtained by each algorithm considered in this experimentation in each evaluation metric. The cases of statistically significant differences according to the Nemenyi test in each evaluation metric are summarized in Table 11.10. Table 11.11 presents the average Friedman ranks computed over the average Friedman ranks of the algorithms on the evaluation metrics. The critical diagram associated with the Nemenyi test corresponding to such Friedman ranks can be seen in Figure 11.4. In Tables 11.9 and 11.11, the best results are marked in bold font and the second-best results in italic font.

Table 11.9: Average Friedman rank obtained by each algorithm of the experimental analysis with label ordering procedures in CC in each evaluation metric.

Metric	BR	CC	ReliefF_CC	CondEnt_CC	ImpCorr_CC
Hamming Loss	**2.1**	3.6	3.1	3.3	*2.9*
Subset Accuracy	4.5	2.9	3.15	**2.2**	*2.25*
Accuracy	4	3.2	3.2	*2.5*	**2.1**
F1	3.9	3.1	3.3	*2.5*	**2**
Micro F1	3.2	3.6	3.1	*2.9*	**2.2**
Macro F1	3.9	3.1	*2.8*	3.2	**2**
One Error	*2.95*	3.15	3.1	**2.55**	3.25
Coverage	4.5	3.2	3.2	*2.1*	**2**
Ranking Loss	4	3.4	3.1	*2.4*	**2.1**
Average Precision	3.7	3.4	3	**2.4**	*2.5*

The following points should be noted about the results:

- In general, the BR algorithm obtains, by far, the worst performance. It is since this algorithm completely ignores the correlations among the labels, and the labels are usually not independent. The only evaluation measure in which BR obtains good results is Hamming Loss, which indicates the symmetric differences between the real label sets and the predicted ones. It is because, as pointed out before, in MLC, there are

Table 11.10: Summary of the results of the Nemenyi tests of the experimental analysis with label ordering procedures in CC in each evaluation metric. In each cell, the algorithm of the column performs significantly better than the method of the row in the evaluation measure indicated in the cell.

	BR	CC	ReliefF_CC	CondEnt_CC	ImpCorr_CC
BR	-			Subset Accuracy, Coverage	Subset Accuracy, Coverage
CC		-			
ReliefF_CC			-		
CondEnt_CC				-	
ImpCorr_CC					-

Table 11.11: Average Friedman ranks computer over the Friedman ranks obtained by the algorithms of the experimental analysis with label ordering procedures in CC in the evaluation metrics.

Algorithm	Average Friedman Rank
BR	4.2
CC	3.8
ReliefF_CC	3.2
CondEnt_CC	2.1
ImpCorr_CC	**1.7**

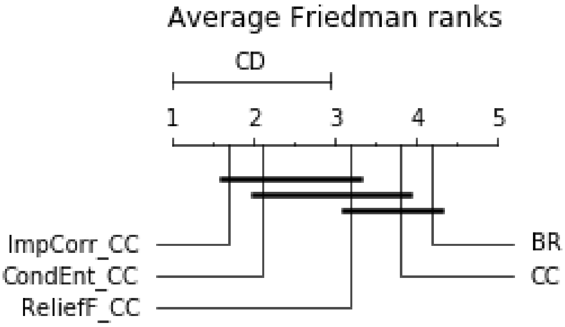

Figure 11.4: Critical diagram corresponding to the Nemenyi test associated with the average Friedman ranks in the experimental analysis with label ordering procedures in CC. CD = Critical Distance.

often very few instances that have associated a certain label. Hence, BR, which does not consider correlations among labels, usually predicts that the labels are irrelevant for the instances. Since BR obtains poor performance in the rest of the classification-based metrics (Accuracy, Subset Accuracy, F1, and Micro and Macro F1), it can be deduced that this algorithm is the least suitable one, among the ones considered here, to handle the class-imbalance problem that frequently appears in MLC. It also achieves quite bad results in ranking-based measures, except for One Error, which only focuses on the top-ranked label.

- The original CC algorithm obtains better results than BR: it obtains a lower average Friedman rank, and it achieves a lower Friedman rank in seven evaluation measures. It is because CC does not assume that the labels are independent, but it considers a label order, and each classifier takes the predictions of the previous ones according to that order into account. In this way, CC captures some label correlations. Nevertheless, CC performs worse than the versions of this algorithm that previously study the correlations among the labels to determine the label order, in both classification-based and ranking-based evaluation measures. The reason is obvious: CC considers a random label order, and the performance of CC is strongly influenced by that label order.

- CC with the label ordering method based on ReliefF performs better than the original CC method; the Friedman rank of the ReliefF_CC method is lower than the one of CC in most of the evaluation metrics, and the average Friedman rank is also lower. It is because ReliefF_CC previously studies the correlations among the labels to order them, unlike CC, which considers a random order. However, ReliefF_CC obtains worse performance than the other two methods based on CC considered here that order the labels by considering the correlations between them. We explained the reasons in Section 11.5.1: it is difficult to establish a good threshold based on the correlation scores to determine the list of labels that are correlated with each one. In addition, in the greedy procedure carried out in ReliefF_CC for ordering the labels, the correlation of each candidate label with the ones already inserted has little influence, which is a shortcoming.

- In consequence, CondEnt_CC and ImpCorr_CC are the algorithms that achieve the best results among the ones considered here; they attain a lower Friedman rank than the other algorithms in most of the evaluation measures, and the average Friedman ranks are pretty lower too. CondEnt_CC and ImpCorr_CC perform significantly better than BR according to the Nemenyi test in both Subset Accuracy and Coverage. These are the only cases of statistically significant differences via this test in the evaluation metrics. Moreover, CondEnt_CC and ImpCorr_CC are not connected with BR via a segment in the critical diagram of Figure 11.4. So, these two algorithms significantly outperform BR according to the Nemenyi test associated with the Friedman test computed over the Friedman ranks in the evaluation metrics. The previous points are since the label ordering methods employed in CondEnt_CC and ImpCorr_CC do not present the problem of the threshold that appears in ReliefF_CC.

- Our proposed ImpCorr_CC method performs better than CondEnt_CC as it obtains a lower average Friedman rank, and the Friedman rank of ImpCorr_CC is lower than the one of CondEnt_CC in seven evaluation metrics, whereas, in the other three evaluation measures, the opposite happens. Furthermore, according to the Nemenyi test associated with the Friedman test computed over the Friedman ranks in the evaluation measures, ImpCorr_CC performs significantly better than CC, unlike CondEnt_CC. It is because, to insert the labels in the chain, our proposal considers, for each candidate label, its correlation with the labels already inserted and its correlation with the labels not inserted yet, whereas CondEnt_CC just considers the correlation of each candidate label with the ones not inserted yet. Furthermore, CondEnt_CC estimates the correlations among the labels via classical probability theory, while our proposal uses the A-NPI-M for this purpose. As explained in Section 11.2.3, imprecise probability models are more appropriate to be employed than precise probabilities in MLC since, in this field, the intrinsic label noise might be higher than the intrinsic class noise in traditional classification.

  Specifically, the difference between the performance of both methods in classification-based evaluation metrics is quite notable: the Friedman rank of CondEnt_CC is slightly lower in Subset Accuracy, whereas ImpCorr_CC obtains a much lower Friedman rank in Accuracy, Hamming Loss, and F1-based metrics. The differences are very notable in Micro and Macro F1. Regarding the ranking-based evaluation metrics, in One Error, ImpCorr_CC performs far worse than CondEnt_CC; it is the only evaluation metric in which our proposal does not achieve good results; in contrast, the Friedman rank of ImpCorr_CC is lower than the one of CondEnt_CC in Ranking Loss; the differences in Coverage and Average Precision are hardly appreciable. Thereby, it can be stated that both algorithms obtain equivalent results in ranking-based evaluation metrics.

To summarize, BR, which ignores the label correlations, obtains the worst results; among the CC-based methods, the original CC algorithm is the one that performs worst; and our proposed label ordering method achieves better results than the ones based on label correlations developed so far, especially in classification-based evaluation metrics.

11.6 Conclusions

Most of the Multi-Label Classification (MLC) methods proposed so far are based on classical probability theory. In this chapter, we have developed new MLC algorithms that use imprecise probability models. We have shown that the intrinsic label noise in MLC tends to be higher than the intrinsic class noise in traditional classification. Hence, as algorithms based on imprecise probabilities have obtained better performance than the ones that employ precise probabilities when there is class noise in the data, our proposed MLC methods are more suitable than the ones developed so far based on classical probability theory. Experimental studies have corroborated this point. Specifically, the main contributions of this chapter can be summarized as follows:

- Firstly, we have analyzed the use of traditional classification algorithms based on imprecise probabilities in problem transformation methods. Remark that algorithms within this category transform the MLC task into multiple traditional classification problems and then combine the solutions of such problems to output a solution for the MLC task. Specifically, we have studied the use of Credal C4.5 (CC4.5) in two problem transformation methods: Binary Relevance (BR) and Calibrated Label Ranking (CLR). BR is a quite simple approach to MLC that has obtained good results in practice, and CLR exploits pairwise label correlations and mitigates the class-imbalance problem that often arises in MLC. We have shown that, as CC4.5 is less sensitive to noise than C4.5, both BR and CLR are more robust to noise in the labels with CC4.5 than with C4.5. Consequently, since the intrinsic label noise in MLC might be higher than in traditional classification, CC4.5 is more suitable than C4.5 to handle the binary classification tasks in BR and CLR. Experimental results have revealed that both BR and CLR achieve better results with CC4.5 than with C4.5, the improvement being more notable as there is more noise in the labels. The difference between the performance of CC4.5 and C4.5 is more notable in metrics based on instances than in label-based metrics. It happens because the proportion of instances with an error in any label may be much higher than the proportion of instances with an error in a specific label.

- We have proposed a new adaptation of Decision Trees for MLC that uses the A-NPI-M for computing the uncertainty-based information about the label set in the nodes for the split criterion and to predict the posterior probabilities about the relevance of the labels for the instances at leaf nodes. It has been shown that our proposed adaptation is more robust to label noise than the one proposed so far, which is based on precise probabilities. An experimental analysis has been carried out with several datasets, MLC evaluation metrics, and noise levels to compare the performance of our proposed adaptation of Decision Trees for MLC and the existing adaptation based on classical probability theory. Such an experimental analysis has highlighted that our proposed adaptation outperforms the one proposed so far, and the improvement is more notable as there is more noise in the labels. Therefore, the A-NPI-M is more suitable than classical probability theory to be employed in the adaptations of Decision Trees for MLC, especially when there is noise in the labels.

- Also, we have developed two new lazy approaches to MLC: Binary Relevance Credal Nearest Neighbors (BR-Credal-NN) and Multi-Label Credal Nearest Neighbors (ML-Credal-NN). As the original Multi-Label Nearest Neighbors (ML-NN) and Binary Relevance Nearest Neighbors (BR-NN) algorithms, to classify a new instance, our proposed methods employ statistical estimators based on the nearest neighbors of the instance. Nonetheless, whereas the original algorithms use relative frequencies with Laplacian correction for the statistical estimators, BR-Credal-NN and ML-Credal-NN use the A-NPI-M. We have shown that our proposed ML-Credal-NN and BR-Credal-NN methods predict that an instance has associated with it a certain label more frequently than the existing ML-NN and BR-NN algorithms, especially when there is noise in the labels. We have carried out an experimental study with many MLC evaluation metrics and different MLC datasets, with and without noise in the labels, to compare the performance of our proposed lazy approaches, based on the A-NPI-M, with some of the existing lazy MLC algorithms based on classical probability theory. Such an experimental study has highlighted that BR-Credal-NN and ML-Credal-NN obtain the best results in Accuracy and F1 measures. Remark that F1 is a well-established evaluation metric in the literature to analyze the performance of algorithms for unbalanced classification problems. In addition, our proposed algorithms are more robust to noise in F1 metrics. In this way, ML-Credal-NN and BR-Credal-NN are more appropriate than ML-NN and BR-NN to tackle the class-imbalance problem that frequently appears in MLC, especially with noise in the labels. In ranking-based metrics, BR-Credal-NN obtains poor results, but ML-Credal-NN does not perform worse than any of the lazy MLC algorithms considered in our experimentation. Thus, ML-Credal-NN is far more suitable than BR-Credal-NN for handling the multi-label ranking problem.

- It must be also remarked that one of the main challenges in MLC is to exploit the correlations between the labels and Classifier Chain (CC) is considered a simple and effective method to exploit label correlations. CC considers a binary classifier per label in which the previous labels according to an established order are employed as additional predictive attributes. Such an order strongly influences the performance of CC, and it is not a trivial question to determine the optimal label order. Most of the label ordering methods proposed so far are based on label correlations. A new label ordering method has been proposed in this chapter. It estimates the correlation between each pair of labels via the A-NPI-M and then orders the labels by means of a greedy procedure. In it, for each candidate label, the average correlation between that label and the ones already inserted is considered, as well as the average correlation between that label and the ones not inserted yet. It has been shown that our proposed procedure presents some advantages over the ones developed so far based on label correlations; it uses imprecise probabilities for estimating label correlations, which is more suitable than employing classical probability theory; it does not utilize a threshold to determine the list of labels that are correlated with each one; our proposal, for each candidate label to insert in the chain, considers the correlation of the labels already inserted with it and the correlation of the labels non-inserted with the candidate label. Experiments have been carried out with several MLC datasets and many MLC evaluation metrics to check the performance of our proposed method. We have compared our proposal with BR, the original CC algorithm, and CC with the label ordering methods proposed so far based on label correlations. Such experiments have revealed that the CC-based algorithms outperform BR, which ignores the label correlations; the original CC method is the one that obtains the worst results among the ones based on CC; as expected, our proposed label ordering procedure achieves better performance than the ones based on label correlations developed so far, the improvement being especially notable in classification-based evaluation metrics.

Part IV

CONCLUSIONS AND FUTURE WORK

12 | CONCLUSIONS AND FUTURE DIRECTIONS

12.1 Concluding remarks

Classical probability theory (PT) is the standard way of representing the information about a finite set of alternatives provided by an expert or dataset. Nonetheless, in many cases, this representation might not be suitable since the available information is not sufficient for precisely determining the probability of each alternative. For this reason, many imprecise probability theories and models have been developed in the literature. Concerning imprecise probability theories, some theories are more general than others. One of the most general ones is the theory based on general credal sets. In addition, as these theories have particular mathematical properties, some theories are more suitable than others in specific situations. Evidence theory (ET) has been widely used in the literature to handle uncertainty-based information, and reachable probability intervals are easy to understand and manage, have high expressive power, and can be efficiently computed. In fact, they have been frequently employed in practical applications such as classification. As demonstrated by Abellán [2], ET does not generalize reachable probability intervals, and the converse is also not satisfied. With regard to the imprecise probability models, the Imprecise Dirichlet Model (IDM), proposed by Walley [209], satisfies some principles that were claimed to be desirable for inference, such as the Representation Invariance Principle (RIP). Nevertheless, this model assumes previous knowledge about the data through a parameter. In practical applications, it has not been possible so far to determine the optimal value of the parameter for each specific case. The Non-Parametric Predictive Inference Model (NPI-M) [58, 59] was developed to solve this shortcoming. The NPI-M is a non-parametric approach that does not make previous assumptions about the data. Even so, the set of probability distributions compatible with the NPI-M is not convex. Indeed, when the NPI-M is employed, it is necessary to deal with difficult constraints. For this reason, in [5], the Approximate Non-Parametric Predictive Inference Model (A-NPI-M) was proposed. The A-NPI-M consists of the convex hull of the set of probability distributions consistent with the NPI-M and belongs to the reachable probability intervals theory. Hence, the

A-NPI-M is more manageable than the NPI-M to be employed in practical applications.

Within imprecise probability theories and models, new tools for representing the available information are needed. Such tools are known as uncertainty measures. The basis of the study of uncertainty measures in imprecise probability theories is the study of uncertainty measures in ET. In such a theory, uncertainty-based information can always be represented via a belief function. The maximum entropy on the set of probability distributions consistent with a belief function is the only uncertainty measure in ET so far that satisfies all essential mathematical properties and behavioral requirements. Also, in the theory based on general credal sets, the maximum entropy is a well-established uncertainty measure as it satisfies the fundamental properties. However, the maximum entropy requires a considerably high computational cost. Actually, the algorithms proposed so far to compute the maximum entropy in ET are notably complex. For this reason, many alternatives to the maximum entropy in ET have been proposed during the last years. A well-known alternative to the maximum entropy is the Deng entropy. In previous works, it was proved that this measure violates most of the fundamental mathematical properties for uncertainty measures in ET, and its behavior in some situations is questionable. Two modifications of this measure were proposed a few years ago for solving some drawbacks of the Deng entropy. It should be remarked that belief intervals for singletons are easier to manage than belief functions for representing the uncertainty-based information in ET. Hence, many alternatives to the maximum entropy in ET proposed during the last years are based on belief intervals for singletons. Furthermore, there is no algorithm so far to compute the maximum entropy on a general credal set. An algorithm for the maximum entropy on Choquet capacities of order 2 was developed [16]. Moreover, Abellán [2] developed algorithms for the computation of the main uncertainty measures with the IDM. An algorithm to compute the maximum entropy with the NPI-M was also developed in [5].

Situations in which it is necessary to represent the information about a finite set of possible alternatives provided by a dataset arise in classification, an essential area within Data Mining. This task consists of predicting, for an instance described via a set of attributes or features, the value of a variable under study called the class variable. In order to make such a prediction, classification algorithms usually need to represent the available information about the class variable given the values of the attributes. One of the most simple classification algorithms is Naïve Bayes (NB). It assumes that all attributes are independent given the class variable. Despite this unrealistic assumption, NB has obtained good results in practice, comparable with more sophisticated classification methods. The estimation of the probabilities plays a crucial role in NB. Many years ago, Cestnik [47] proposed a new NB model that takes the prior probabilities into account for the estimation of the conditional probabilities. Such a model has obtained better performance that the NB models based on classical estimators. Also, Decision Trees are very simple, transparent, and interpretable models. Within Decision Trees, C4.5 is a well-known classification algorithm. Classical Decision Trees use uncertainty measures based on PT for selecting the attribute to split in each node. Decision Trees that utilize uncertainty measures on credal sets to represent the uncertainty-based information about the class variable at each node, known as Credal Decision Trees (CDTs), were developed a few years ago. Within CDTs, a version of C4.5 based on imprecise probabilities, called the Credal C4.5 algorithm (CC4.5), was proposed. CDTs and classical Decision Trees obtain statistically equivalent results without noise in the data and CDTs perform significantly better than classical Decision Trees when classifying class noise. Moreover, ensembles of classifiers often achieve better results than individual classifiers. They consider multiple individual classifiers and combine their predictions to give a final prediction. The key point for the success of an ensemble scheme is that the base classifiers are not only accurate but also diverse. Therefore, Decision Trees are appropriate for ensembles as, in these classifiers, little variations in the training set may lead to considerable variations in the learned model.

Most classification algorithms aim to minimize the number of misclassified instances. This would be optimal if all classification errors had the same importance. Nonetheless, in practical applications, classification errors tend to yield different costs. For this reason, many classifiers that take the error costs into account, known as cost-sensitive classifiers, have been developed in the last years.

Cost-insensitive and cost-sensitive classifiers usually predict a single value of the class variable when classifying an instance. However, in some cases, classifiers predict a set of class values because there is not sufficient available information to point out a unique value of the class variable. This is known as Imprecise Classification. The performance of an imprecise classifier is evaluated via its trade-off between accuracy (the real class value is among the predicted ones) and informativeness, which is measured by the average number of predicted class values. The first Imprecise Classification method was the Naïve Credal Classifier (NCC), which combines the IDM with the naïve assumption to output imprecise predictions. Afterwards, the first Imprecise Classification algorithm based on Decision Trees was developed, which is called the Imprecise Credal Decision Tree (ICDT). ICDT obtained significantly better performance than NCC as the former model is much more informative than the latter. Both NCC and ICDT were adapted for cost-sensitive scenarios.

Traditional classification assumes that each instance has a unique value of a class variable. In some domains, Multi-Label Classification (MLC) fits better than traditional classification since each instance might belong to multiple labels simultaneously. MLC aims to predict the set of labels associated with an instance. Many MLC algorithms have been proposed so far. They can be divided into two groups. On the one hand, the problem transformation methods convert the MLC task into multiple traditional classification problems and then combine their solutions to give a prediction for the MLC problem. On the other hand, the algorithm adaptation methods directly adapt the existing algorithms for traditional classification to MLC. Most of the algorithms proposed so far for MLC use PT. It should be remarked that, as the number of labels in MLC tends to be very high, one of the main challenges of MLC is to exploit label correlations. The Classifier Chain algorithm (CC) is considered a simple and effective method to exploit label correlations in MLC. CC considers a binary classification problem per label in which the previous labels according to an established order are used as additional predictive attributes. The performance of CC is strongly influenced by the label order, and there is no way so far of determining the optimal label order. Many label ordering methods in CC have been developed so far. Most of them estimate label correlations via classical PT before ordering the labels. It must also be remarked that, in MLC, there are often very few instances that belong to a certain label. Consequently, MLC algorithms tend to suffer from a class-imbalance problem.

In this thesis work, we have analyzed some imprecise probability theories and models; we have also made a critical analysis of some uncertainty measures in ET, and we have proposed an uncertainty measure on belief intervals for singletons that satisfies all essential mathematical properties and behavioral requirements; we have proposed new classification algorithms based on imprecise probability models, including the above-commented special types of classification: Imprecise Classification and MLC, that outperform the ones of the state-of-the-art.

In concrete, we list below the main contributions of this thesis work:

- Credal sets representable by belief functions and reachable probability intervals have been characterized: we have provided a set of necessary and sufficient conditions that a given reachable set of probability intervals must satisfy to be representable by means of a belief function. It has been demonstrated that, in order to check such conditions, it is required to consider several subsets and check some simple inequalities with the sums of the lower and upper probabilities on such subsets. The subsets are also simple and fast to compute. We have also given a characterization of belief functions representable via reachable probability intervals. Specifically, it has been demonstrated that the necessary and sufficient condition for a belief function to be representable through a reachable set of probability intervals is the following one: the difference between any pair of non-singleton focal elements of the corresponding basic probability assignment (BPA) has a cardinality lower or equal than one. By employing our given condition, we have characterized some special types of belief functions, such as p-boxes or necessity measures, that can be represented via reachable sets of probability intervals.

- With regard to imprecise probability models, we have analyzed the main properties of A-NPI-M credal sets, comparing them with IDM credal sets. It has been shown that, as with the IDM, as long as the sample size converges to infinity, A-NPI-M credal sets converge to a unique probability distribution, computed through relative frequencies; the A-NPI-M is a more imprecise model than the IDM with the most utilized value of the parameter, the one recommended in the literature; one of the most remarkable properties of A-NPI-M credal sets is that they cannot always be represented through a belief function, unlike IDM credal sets. The calculation of the Möbius inverse for the A-NPI-M is more complex than for the IDM. The same occurs with the set of extreme points of the credal set. Therefore, the A-NPI-M is a more complex model than the IDM. Nonetheless, it must be remarked that the IDM assumes previous knowledge about the data via a parameter, unlike the A-NPI-M.

- Concerning uncertainty measures, we have made a critical analysis of two modifications of the Deng entropy proposed a few years ago. It has been proved that such modifications, similar to the original Deng entropy, do not satisfy most of the crucial mathematical properties for uncertainty measures in ET, and their behavior in some scenarios is also questionable. Furthermore, we have carried out a study about the essential mathematical properties and behavioral requirements for total uncertainty measures on belief intervals for singletons. That study has been based on the one previously carried out for total uncertainty measures on BPAs. We have shown that none of the uncertainty measures on belief intervals for singletons proposed so far verifies all the fundamental mathematical properties and behaviors for this type of measure. We have also proposed a total uncertainty measure on belief intervals for singletons, which consists of the maximum entropy on the credal set associated with such intervals. We have proved that, even though our proposed measure requires a more complex computation than the other uncertainty measures on belief intervals for singletons proposed so far, it is the only one that satisfies all the crucial mathematical properties and behavioral requirements for uncertainty measures on belief intervals for singletons. We have also highlighted that our proposal gives an upper bound of the maximum entropy on the credal set compatible with a BPA, the well-established uncertainty measure in ET, the computation of the former measure being notably faster than the latter. Moreover, we have shown how to compute the most important uncertainty measures on A-NPI-M credal sets. Such procedures represent useful tools to make the A-NPI-M very suitable for practical applications.

- Within traditional classification, we have presented a new version of the NB algorithm, called the Imprecise m-probability estimation Naïve Bayes (ImNB), that considers the prior probabilities of the class values to estimate the conditional probabilities, as the Cestnik approach. However, ImNB uses the well-established uncertainty measure on credal sets for estimating the prior probabilities, unlike the Cestnik model, which employs relative frequencies with Laplacian correction to estimate such probabilities. Thereby, our proposed ImNB is more robust to noise than the Cestnik model. An experimental study with several noise levels has highlighted that ImNB performs better than the Cestnik approach and the versions of NB that use classical estimators of the probabilities, with and without noise in the data.

- Improvements over the Imprecise Classification algorithms developed so far have been proposed. Specifically, we can summarize the contributions of this thesis work related to Imprecise Classification in the following issues:

    - We have proposed a new version of the ICDT algorithm that employs the A-NPI-M for the split criterion and the probability intervals at leaf nodes (ICDT-ANPI), whereas the existing ICDT uses the IDM. Experimental results have shown that ICDT-ANPI performs equivalently to ICDT with the best choice of the IDM parameter. Consequently, the A-NPI-M is more appropriate than the IDM for Decision Trees for Imprecise Classification since the former model does not assume previous knowledge about the data via a parameter, unlike the latter model.

- A new version of the NCC algorithm has been developed, called the Extreme Prior Naïve Credal Classifier (EP-NCC). Unlike NCC, EP-NCC takes the lower and upper prior probabilities of the class values into account to estimate the lower and upper conditional probabilities. It has been shown that the predictions made by EP-NCC are probably more informative than the predictions made by NCC, the risk of incorrect predictions not being much higher with EP-NCC. An experimental analysis has revealed that EP-NCC performs significantly better than NCC as the former method is much more informative than the latter, while the difference between the performance of both algorithms in accuracy is not statistically significant. Such an experimental analysis has also highlighted that EP-NCC and ICDT obtain equivalent performance, but ICDT requires a much higher computational time than EP-NCC. Hence, due to its good performance and low computational time, EP-NCC is more suitable for large datasets for Imprecise Classification than the existing algorithms for such a task. This is an important point in favor of our proposed EP-NCC algorithm because of the increasing amount of data in every area.

- The first ensemble method for Imprecise Classification has been presented in this thesis work. It has been taken into account that the Bagging scheme has obtained good performance in precise classification, especially when it has been used with CDTs, which encourage diversity. Thus, our proposed ensemble method for Imprecise Classification consists of a Bagging scheme using the ICDT algorithm (the adaptation of CDT for Imprecise Classification) as the base classifier (Bagging-ICDT). The key point is how to combine the predictions made by multiple imprecise classifiers. As commented before, it is not a trivial question because, if the imprecise predictions are not suitably combined, then the ensemble may not perform better than an individual imprecise classifier since an excessive reduction of the information can be produced. Our proposed combination technique aims that the Bagging imprecise classifier is as informative as possible. Such a technique consists of predicting as non-dominated only the class values with the lowest possible level of dominance, which implies that it is not very conservative. Via an experimental analysis, we have shown that Bagging-ICDT with our proposed combination technique performs much better than the ICDT method; Bagging-ICDT is far more informative than

  ICDT, whereas the difference between the performance of both algorithms in making correct predictions is not significant.

  - Concerning cost-sensitive classification, we have proposed a new cost-sensitive Imprecise Credal Decision Tree that weights the instances by considering the cost of misclassifying the corresponding class value. Our proposed method takes the error costs into account in the tree-building process, unlike the existing cost-sensitive Imprecise Credal Decision Tree, which just considers the misclassification costs to classify instances at leaf nodes. We have shown that the criterion used by our proposed cost-sensitive Imprecise Credal Decision Tree for classifying instances at leaf nodes might be more effective than the one employed by the existing cost-sensitive Imprecise Credal Decision Tree because the predictions made may be more informative. An experimental study has highlighted that our proposed cost-sensitive Imprecise Credal Decision Tree significantly outperforms the existing one; although the misclassification cost of our proposed method is higher, it is much more informative and achieves a better trade-off between low cost of incorrect classifications and informative predictions. In this way, we conclude that our proposed cost-sensitive Imprecise Credal Decision Tree is more appropriate than the existing one for practical applications where the error costs are different and the available information is not enough for classifiers to predict a single class value.

- We have also proposed new MLC algorithms based on imprecise probability models. We have shown that the intrinsic label noise in MLC is probably higher than the intrinsic class noise in traditional classification. In consequence, since algorithms that use imprecise probabilities perform better than the ones based on classical PT when there is class noise in the data, our proposed MLC methods might be more suitable than the ones developed so far based on precise probabilities. We have checked this issue via experimental studies. In concrete, the contributions of this thesis work regarding MLC can be summarized in the points below:

  - Firstly, we have analyzed the use of CC4.5 in two problem transformation methods: Binary Relevance (BR) and Calibrated Label Ranking (CLR). BR is a very simple MLC method that has obtained good performance in practice, and CLR exploits pairwise label correlations and alleviates the class-imbalance problem that tends to appear in MLC. We have shown that, as CC4.5 is more robust to class noise than C4.5, both BR and CLR are less sensitive to noise in the labels with CC4.5 than with C4.5. Hence, since the intrinsic label noise in MLC may be higher than the intrinsic class noise in traditional classification, CC4.5 is probably more suitable than C4.5 to tackle the binary classification problems in BR and CLR. Experimental results have shown that both BR and CLR obtain better performance with CC4.5 than with C4.5, the improvement being more significant as the noise in the labels is higher.

- We have proposed a new adaptation of Decision Trees for MLC that employs the A-NPI-M for the split criterion and to predict the posterior probabilities about the relevance of the labels for the instances at leaf nodes. We have shown that our proposed adaptation is less sensitive to label noise than the one proposed so far, which is based on classical PT. Experimental results have highlighted that our proposed adaptation performs better than the one proposed so far, the improvement being more notable as there is more noise in the labels. Therefore, the A-NPI-M is more appropriate than classical PT to be used in the adaptations of Decision Trees for MLC, especially when there is noise in the labels.
- Also, we have presented two lazy MLC algorithms that, in order to classify an instance, employ statistical estimators based on the neighboring instances, similar to some existing lazy MLC methods. Nevertheless, our proposed lazy methods use the A-NPI-M for such statistical estimators, unlike the existing ones, which utilize relative frequencies with Laplacian correction. We have shown that our proposed lazy MLC algorithms predict that a label is relevant for an instance more frequently than the existing ones, and this issue is enhanced with noise in the labels. An experimental study has revealed that our proposed lazy MLC methods significantly outperform the ones proposed so far in F1-based evaluation metrics. Thus, as F1 is a well-established evaluation metric to analyze the performance of algorithms for unbalanced classification problems, it can be stated that our proposed lazy MLC algorithms, based on the A-NPI-M, are more suitable than the existing ones based on precise probabilities to handle the class-imbalance problem that usually arises in MLC, especially with noise in the labels.
- Finally, we have proposed a new label ordering procedure in CC that estimates the correlation between each pair of labels via the A-NPI-M and then orders the labels through a greedy procedure. In such a procedure, for each candidate label, the average correla-

tion between that label and the ones already inserted is considered, as well as the average correlation between that label and the ones not inserted yet. It has been shown that our proposed procedure presents some advantages over the ones developed so far based on label correlations; it employs an imprecise probability model to estimate label correlations, which is more appropriate than precise probabilities; our proposed method, for each candidate label, takes into account the correlation of the labels already inserted with it and the correlation of the labels non-inserted with the candidate label, while some of the label ordering methods proposed so far only consider the correlations between the candidate label and the ones not inserted yet. An experimental study has shown that our proposed label ordering method performs better than the ones based on label correlations developed so far.

12.2 Future research

The management of uncertainty-based information in some imprecise probability theories is still an open research line. Such management is very important in some crucial tasks, such as classification. Within this task, including Imprecise Classification and MLC, new algorithms based on imprecise probability models that perform better than the ones developed in this thesis work could also be proposed. Moreover, the tools presented here could be employed in practical applications to extract useful information.

We list below some specific ideas for future research:

- Some of the essential mathematical properties for uncertainty measures on BPAs are debatable. For example, in ET, there more are types of uncertainty than in classical PT: conflict and non-specificity coexist in ET, while the only type of uncertainty existing in PT is conflict. Consequently, it may be logical that the range of an uncertainty measure on BPAs is larger. Hence, the set of crucial mathematical properties for uncertainty measures on BPAs could be revised. The same happens with the set of fundamental mathematical properties for uncertainty measures on belief intervals for singletons proposed in this thesis work.

- Few Imprecise Classification methods have been developed so far. For future work, many of the traditional classification algorithms based on classical PT could be adapted for Imprecise Classification by employing imprecise probability theories or models.

- It must be noted that the first ensemble method for Imprecise Classification has been presented in this thesis work. In this way, other ensemble schemes that have obtained good performance in standard classification, such as Boosting or Random Forest, could be adapted for Imprecise Classification. Moreover, as said previously, the combination of multiple imprecise predictions is not trivial. The combination technique of our proposed ensemble algorithm for Imprecise Classification tries that the ensemble is as informative as possible, although it assumes a higher risk of incorrect predictions. This technique has obtained good experimental results, but it might not be optimal. Therefore, it would be interesting to study other techniques for combining the predictions made by multiple imprecise classifiers.

- With regard to cost-sensitive Imprecise Classification, other cost-sensitive Imprecise Credal Decision Trees could be developed by considering other ways of using the instance weights for the split criterion or via other criteria to make the predictions at leaf nodes. In addition, it would be worth proposing ensemble methods for cost-sensitive Imprecise Classification that use our proposed cost-sensitive Imprecise Credal Decision Tree as the base classifier.

- New MLC algorithms based on imprecise probabilities have been proposed in this thesis work. They have obtained better performance than the existing ones that use classical PT. For future work, this motivates us to develop new MLC methods based on imprecise probability models that outperform the ones of the state-of-the-art. Also, imprecise probability models could be utilized in ensembles of Decision Trees for MLC. Furthermore, it would be interesting to propose new methods for exploiting label correlations in MLC based on imprecise probabilities. For example, in CC (a simple and effective method for exploiting label correlations in MLC), other label ordering methods based on imprecise probability models that achieve better results than the one developed in this thesis work could be proposed.

- It would be worth developing algorithms that combine MLC with Imprecise Classification. Even though this point has not been covered in this thesis work, in the literature, there are some works about methods that output multi-label imprecise predictions. Examples can be found in [45, 46, 168].

- Finally, the classification algorithms based on imprecise probability models presented here could be used in practical applications, such as *credit* *scoring*, *medical diagnosis*, *software defect prediction*, *traffic accident analysis*, *text categorization*, or *biology*, to extract useful knowledge in these domains.

Part V

APPENDIX

A PRACTICAL APPLICATIONS OF IMPRECISE PROBABILITY MODELS

On the one hand, an estimated 1.27 million people die, and 20-50 million people are injured in traffic accidents every year, with devastating human and economic impact [219]. For this reason, it is fundamental to analyze the main causes of the serious severity in traffic accidents to avoid fatal injuries. In this way, *traffic accident analysis* is an essential area nowadays. Within this area, data mining techniques are commonly applied to traffic datasets for extracting the main causes of fatalities in traffic accidents.

On the other hand, the *credit risk analysis* is a crucial issue for banks and financial institutions. It consists of applying techniques to credit databases to extract information about when a credit should be given to a client depending on a set of features of such a client. It is important to remark that, in credit risk, any small improvement might lead to considerable benefits [151].

In this appendix, we apply imprecise probability models to techniques for traffic accident analysis (Section A.1) and credit risk analysis (Section A.2)

A.1 Traffic accidents of novice drivers in urban areas

A.1.1 Introduction

Many accidents occur in urban areas [196]. In addition, the factors that affect road accident severity in urban and no urban areas are different [198]. Also, many works in the literature, such as [206, 215] have revealed that driving experience is an important factor in accident analysis, since inexperienced drivers are more likely to suffer or cause fatal injuries. For instance, in [98], it was concluded that adolescents, inexperienced in most cases, are not really worried about certain dangers, and that this lack of experience is the reason of many crashes. In fact, novice drivers have a different perception about the risk [35, 131], and novice drivers have less visual attention than the experienced ones [204, 205].

In this section, which corresponds to our published work [164], we study accidents in urban areas in Spain involving drivers with 3 or fewer years of driving experience. In particular, our goal is to analyze the main causes of fatal injuries in this kind of accidents. This will help road safety analysts and managers to identify the main problems related to inexperienced drivers and take measures for trying to reduce the number of accidents of this type and alleviate their consequences. Accidents in intersections are analyzed separately.

A.1.2 Data and Methods

A.1.2.1 *Accident data*

The Spanish General Traffic Directorate (DGT) collected accident data over a period of 5 years (2011-2015). These data contain three tables per year. One of these tables (accidents) refers to the characteristics of the crashes (weather conditions, time, weekday . . . , and so on). The second table (people) contains data about the people involved in each accident. Finally, the third table (vehicles) covers the information of the vehicles involved in each crash. A description of the meaning of possible variable values for these tables can be found in `https://sedeapl.dgt.gob.es`.

The afore-mentioned data has been pre-processed to get the data in a suitable format for the data mining methods employed to extract knowledge from the data. Specifically, some variables that are clearly not relevant have not been considered; we have grouped some attribute values because variables with many possible values often have a negative impact on the performance of data mining algorithms; also, some continuous variables have been discretized. Most of these transformations have been extracted from the work of [169]. The most important transformation is the creation of the variable *driver_type*, which indicates whether the driver has three or fewer years of experience. Since we want to study the impact of driving inexperience on the fatality of accidents in urban areas, only data corresponding to accidents in urban areas and drivers with three years of experience or less are selected.

A.1.2.2 *Information Root Node Variation*

Decision Trees are well-known classification methods as they are very simple, transparent, and interpretable models. One important advantage of DTs is that decision rules (DRs) can be extracted easily. A DR is a logical conditional structure written as an “IF A THEN B" statement, where A is the antecedent and B is the consequent of the rule. In our case, the antecedent is the set of values of several attributes, and the consequent is a class value. Each rule starts at the root node, where the conditioned structure (IF) begins. Each variable that appears in the path represents an IF condition of a rule, that ends in leaf nodes with a THEN value, associated with the class value associated with the leaf node.

However, rules that can be obtained from a single decision tree strongly depend on the root node. Thus, within a single Decision Tree, it is only possible to extract knowledge in the sense indicated by root variable. For this reason, the Information Root Node Variation method (IRNV), proposed in [19], varies the root node to generate different DTs. Therefore, as multiple trees with different structure are considered, the number of rules may be much larger than the number of rules generated using a single Decision Tree.

Furthermore, the split criterion is probably the most important point for building a Decision Tree since different split criteria might lead to considerably different trees. Consequently, the IRNV method also employs several split criteria for the tree-building process.

A.1.2.3 *Selection of the best rules*

A priori, a decision rule can be obtained for each leaf node in a Decision Tree. Nevertheless, in a considerable number of cases, some of these rules are not significant because a rule should not necessarily represent a large number of instances of the dataset. Rules of this kind do not provide useful information for defining safety measures.

For this reason, sufficiently significant rules are extracted using a mechanism based on two parameters:

- **Support** (S): Let us consider a rule of type 'IF A THEN B' ($A \rightarrow B$). Support is defined as the fraction of the data set where A and B are present. In other words, it is the probability that both the antecedent and the consequent occur.

- **Probability** (Pr): It is the probability that the consequent is present given that the antecedent is present. If we have a rule $A \rightarrow B$, then $Pr = P(B|A) = \frac{P(A,B)}{P(A)}$, where $P(A, B)$ is the probability of $A \cap B$.

The selection method consists in choosing only the rules in the set which meet a minimum value (threshold) for both parameters. As argued in [19], the more suitable threshold values for Support and Probability depend on some characteristics, such as the nature of the data (balanced or unbalanced), the interest in the minority class, and the data sample.

A.1.2.4 *Procedure*

In accordance with the research carried out in [9], three split criteria have been considered: The Information Gain Ratio (IGR), the Imprecise Information Gain based on the IDM (IIG-IDM), and the Imprecise Information Gain based on the A-NPI-M (IIG-ANPI).

The software used to build the Decision Trees was Weka. We added the necessary methods to build decision trees using the IRNV with the above-mentioned split criteria. Consistently with other works in the literature, such as [9, 19, 169], we built the Decision Trees with only four levels to get rules which could be useful, simple and easy to understand for road safety analysts. We did not use pruning to build the Decision Trees. The rest of the parameters had the default values used in Weka. We also used the treatment for missing values given by default in Weka.

Our two datasets are relatively small (slightly over 30,000 instances) and the nature of our data is clearly unbalanced (less than the 10% of the instances have fatal severity), as the data set considered in [160]. Similarly, we are strongly interested in extracting rules where the consequent is fatal injury. For this reason, we selected the same Support and Probability thresholds used in [160]: 10% for Pr and 0.1% for S.

A.1.3 Results

The obtained results can be seen in the rules of Tables 2 and 3 of [164]. From these results, we can make the following comments:

- In most of the rules with the highest values of support and probability, part of the antecedent is that the type of accident is running over a pedestrian. Hence, collisions with pedestrians is the type of accident of inexperienced drivers that causes the most fatal injuries in urban areas. For this reason, road safety analysts should urge inexperienced drivers to exercise extreme caution to prevent pedestrian accidents while driving.

- We have also observed that in most of the accidents with fatal severity involving inexperienced drivers the speed of the vehicle was excessive. In fact, in most of the rules with the highest values of support and probability, part of the antecedent is that the infraction of the driver is a high speed. Therefore, speed moderation is crucial for novice drivers, especially during the first few years. As this is a very common infraction, advertising campaigns should be run to raise awareness about this issue among inexperienced drivers.

- The issues mentioned above often give rise to fatal severity in accidents, even when external conditions are good. However, according to part of the antecedents of some rules, the situation could be aggravated under certain circumstances. These circumstances can be the hour, the weekday and, most importantly, the lack of pavements. This factor can lead to fatal injuries, especially in intersections. In consequence, apart from warning novice drivers to moderate the speed in areas without pavements, building more pavements in urban areas (especially in intersections) is desirable.
- Part of the antecedent in some rules corresponding to intersections is that the driver overlooked traffic officer directions. So, another point to consider is that novice drivers must obey traffic officer directions in intersections. They should be aware that the directions given by a traffic officer have priority over the rest of signals and general rules. Thus, advertising campaigns should be run to raise awareness about this point among inexperienced drivers.

A.2 Feature selection in Bayesian networks for credit scoring

A.2.1 Introduction

Many data mining techniques have been employed in the literature for credit risk analysis. Among them, we can mention Bayesian Networks [172]. These models are very interpretable since they use graphical structures to represent dependence relations among the features of the problem. They have been successfully used to work with credit scoring datasets [28, 136, 146, 238]. Indeed, a BN can be used as a classifier but it is not one of the principal virtues of this model. BNs have different characteristics than standard classifiers:

- BNs are interpretable probabilistic models, whereas some classifiers with a high predictive performance perform as black-boxes;
- If the classifier is also interpretable as BNs (for example, a decision tree), then we need to know the values of all the features associated with the case to predict (all the values of the antecedent to know the consequent

in a rule generated by a decision tree). With a BN, we can do inference regardless of the number of observations about the features that we have. Furthermore, BNs are capable of informing about the probability of each value from any feature. These probabilities change when we know the values of any other features. For example, knowing part of the credit applicant data, we will be able to calculate the probability that the credit is positive through inference methods.

- With a BN, we can do inference from causes to effects and from effects to causes, whereas, with another classifier, we can only predict the class variable (causes to effects). Knowing the values of some features, with a BN, we can find the most probable combination of the rest of the features. For example, suppose that a credit is negative and the client is under twenty years of age. Then, we can find the most probable combination of values for the rest of the client features.

The reduction of the number of features can improve the performance and reduce the complexity of a NB. For that aim it is important that the procedure used to select variables could find the most informative features. A higher number of features does not necessarily imply that the learned BN be a better representation of the data available. If we have irrelevant features, the BN could use them and build a model with erroneous relations. Redundant variables will usually deteriorate the goodness of a fitted model. Furthermore, the models including a great number of features become less interpretable because the network is bigger and more complex. For these reasons, it is appropriate to take advantage of a good feature selection algorithm that would remove any irrelevant/redundant variables before learning the network.

One of the most successful feature selection algorithms employed in the literature is the *Correlation-Based Feature Selection* method (CFS) [106]. It consists of a greedy procedure that selects the set of attributes correlated with the class variable by taking into account, for each candidate attribute to insert, the correlations between the candidate attribute with the attributes already inserted and the correlation between the candidate attribute with the class variable. Before the explained greedy procedure, CFS estimated the correlation between each pair of attributes by utilizing classical probability theory.

As pointed out previously, the use of imprecise probabilities has several advantages. The most important of them might be the suitable management of the little reliable information, when the sample size is not enough or there are noisy data. In particular, the Imprecise Information Gain measure (IIG), used for the split criterion in Decision Trees (see Equation (4.22)), has been successfully employed in such models for studying the correlation of an attribute with the class variable, especially with noisy data.

In this section, corresponding to our work [121], we define a new feature selection method to select a subset of informative features. This method will be based on the IIG measure in a forward way to add features. The new feature subset selection algorithm will be called the *Forward Feature Selection based on Imprecise Information Gain* (FFSIIG). Our principal aim is to show that if we build a BN from data using the FFSIIG in a previous step, then we obtain a better representation of the data than the BN built with no previous subset feature selection. Moreover, we will also show that the BN built with the features selected by the FFSIIG is also more representative of the data than a similar BN built with CFS, one of the most used and successful procedures to select variables.

A.2.2 Methodology

A.2.2.1 *Bayesian Networks*

A Bayesian Network (BN) [172] is a graphical model which encodes a joint probability distribution, being composed of a qualitative part, a directed acyclic graph which represents the dependencies among the variables, and a quantitative part, a collection of numerical parameters, commonly conditional probability tables.

The common practice is to learn a BN automatically from a dataset, although it can be built manually from an expert. There are also mixed methods to build a Bayesian Network where the network can be learned automatically from data and manually refined by an expert. In this sense, the problem of learning automatically a BN from data is to find the network that, in some sense, best represents the data.

Once we have obtained a BN, we usually need to determine various probabilities of interest as we get new information or evidence. For example, in a credit scoring problem, we want to know the probability of grant a credit given the data of a new client. Thus, we can define the probability propagation or probabilistic inference [172, 190] as the computation needed to obtain the posterior probability of one or several variables (e.g., grant a credit or not) given the values of other variables (e.g., new client data).

A.2.2.2 *New feature selection method*

Our proposed Forward Feature Selection based on Imprecise Information Gain method (FFSIIG) uses the maximum entropy on IDM credal sets to estimate the gain of information of the class variable when we have a set of attributes.

Hence, in order to evaluate the goodness of a subset of attributes, FFSIIG considers the information gain of the class variable given such attributes, where the uncertainty-based information about the class variable is represented via the maximum entropy on the corresponding IDM credal set.

FFSIIG uses a greedy procedure to select the attributes correlated with the class variable. At each step of such a procedure, among the attributes not selected yet, the one that gives rise to the maximum gain of information of the class variable given the new subset resulting from adding the attribute.

A.2.2.3 *Procedure*

- We have utilized five well-known credit scoring datasets used in other works in the literature such as [9, 151]. A detailed description about these datasets can be seen in Table 2 of [121]. To work with BNs, these datasets have been discretized by using Fayyad and Irani discretization method [88]. After that discretization process, a few continuous variables have been discretized into a single value and, consequently, such variables have been removed.

- For each one of these discretized datasets, we have applied two feature selectors: CFS and FFSIIG. The CFS algorithm was already implemented in Weka. We have added the necessary structured and methods for FFSIIG. The IDM parameter was set to $s = 1$ because it is one of the recommended by Walley [209] and requires a low computational cost.

- After the preprocessing stage, we have used the Elvira System **[elvira]** to build BNs via different methods. Three different approaches have been utilized to learn the structure of the BNs: (1) the score-based K2 algorithm [61]; (2) a local search approach with the BDeu metric [42]; (3) the PC algorithm [193]. The performance measure used is the Kullback-Leibler divergence [132], defined as the distance between the joint probability distributions associated with a candidate network and with the available data set. This measure is accepted as a standard measure of error in the Bayesian networks literature [112, 172].

A.2.3 Comments on the results

The obtained results are shown in Tables 3-6 of [121]. The following points should be remarked about these results:

- For the three learning methods considered here, the BNs represent the data much better when we apply a previous feature selection (CFS or FFSIIG). In fact, for all datasets, the KL divergence to the original data is considerably higher for the BNs learned with all features. Therefore, we can observe that the redundant and/or irrelevant features yields a less representative model of the data.

- Comparing the two best feature selection methods, the KL divergence is lower of the BNs learned with the features selected by FFSIIG than the divergence of the BNs learned after selecting features via CFS for all datasets, except for the 'German' dataset. So, in general, the results obtained with FFSIIG are better than those obtained with CFS. FFSIIG is better in 4 out of 5 datasets. Moreover, for the German dataset, where the KL divergence is lower for the CFS, the difference is not as significant as in the rest of the datasets, where the KL divergences for the CFS are considerably higher than the ones obtained with the FFSIIG.

- Furthermore, the number of features selected by FFSIIG is generally far lower than the number of attributes selected by CFS. In consequence, the BNs obtained with FFSIIG, as a previous step, are more simple and explicative because they are built with a lower number of features than the ones built previously using the CFS method.

BIBLIOGRAPHY

[1] Joaquin Abellan and Serafin Moral. "Maximum of Entropy for Credal Sets". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 11.5 (2003), pp. 587–597.
DOI: 10.1142/S021848850300234X.

[2] Joaquín Abellán. "Uncertainty measures on probability intervals from the imprecise Dirichlet model". In: *International Journal of General Systems* 35.5 (2006), pp. 509–528.
DOI: 10.1080/03081070600687643.

[3] Joaquín Abellán. "Ensembles of decision trees based on imprecise probabilities and uncertainty measures". In: *Information Fusion* 14.4 (2013), pp. 423–430.

[4] Joaquín Abellán. "Analyzing properties of Deng entropy in the theory of evidence". In: *Chaos, Solitons and Fractals* 95 (2017), pp. 195–199. ISSN: 0960-0779.
DOI: 10.1016/j.chaos.2016.12.024.

[5] Joaquín Abellán, Rebecca M. Baker, and Frank P.A. Coolen. "Maximising entropy on the nonparametric predictive inference model for multinomial data". In: *European Journal of Operational Research* 212.1 (2011), pp. 112–122. ISSN: 0377-2217.
DOI: 10.1016/j.ejor.2011.01.020.

[6] Joaquín Abellán, Rebecca M. Baker, Frank P.A. Coolen, Richard J. Crossman, and Andrés R. Masegosa. "Classification with decision trees from a nonparametric predictive inference perspective". In: *Computational Statistics & Data Analysis* 71 (2014), pp. 789–802. ISSN: 0167-9473.
DOI: 10.1016/j.csda.2013.02.009.

[7] Joaquín Abellán and Éloi Bossé. "Drawbacks of Uncertainty Measures Based on the Pignistic Transformation". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48.3 (2018), pp. 382–388.
DOI: 10.1109/TSMC.2016.2597267.

[8] Joaquín Abellán and Javier G. Castellano. "A comparative study on base classifiers in ensemble methods for credit scoring". In: *Expert Systems with Applications* 73 (2017), pp. 1–10. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2016.12.020.

[9] Joaquín Abellán, Griselda López, Laura Garach, and Javier G. Castellano. "Extraction of decision rules via imprecise probabilities". In: *International Journal of General Systems.* 46.4 (2017), pp. 313–331. DOI: 10.1080/03081079.2017.1312359.

[10] Joaquín Abellán and Andrés R. Masegosa. "Imprecise Classification with Credal Decision Trees". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 20.05 (2012), pp. 763–787. DOI: 10.1142/S0218488512500353.

[11] Joaquín Abellán and AndrésR. Masegosa. "An Experimental Study about Simple Decision Trees for Bagging Ensemble on Datasets with Classification Noise". In: *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*. Vol. 5590. Lecture Notes in Computer Science. Springer, 2009, pp. 446–456. ISBN: 978-3-642-02905-9. DOI: 10.1007/978-3-642-02906-6_39.

[12] Joaquín Abellán and Serafín Moral. "A non-specificity measure for convex sets of probability distributions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 08.03 (2000), pp. 357–367. DOI: 10.1142/S0218488500000253.

[13] Joaquín Abellán and Serafín Moral. "Building classification trees using the total uncertainty criterion". In: *International Journal of Intelligent Systems* 18.12 (2003), pp. 1215–1225. ISSN: 1098-111X. DOI: 10.1002/int.10143.

[14] Joaquín Abellán and Serafín Moral. "Difference of entropies as a nonspecificity function on credal sets". In: *International Journal of General Systems* 34.3 (2005), pp. 201–214. DOI: 10.1080/03081070500108609.

[15] Joaquín Abellán and Serafín Moral. "Difference of entropies as a nonspecificity function on credal sets". In: *International Journal of General Systems* 34.3 (2005), pp. 201–214. DOI: 10.1080/03081070500108609.

[16] Joaquín Abellán and Serafín Moral. "Upper entropy of credal sets. Applications to credal classification". In: *International Journal of Approximate Reasoning* 39.2–3 (2005). Imprecise Probabilities and Their Applications, pp. 235–255. ISSN: 0888-613X. DOI: 10.1016/j.ijar.2004.10.001.

[17] Joaquın Abellán. “Equivalence relations among dominance concepts on probability intervals and general credal sets”. In: *International Journal of General Systems* 41.2 (2012), pp. 109–122.
DOI: 10.1080/03081079.2011.607449.

[18] Joaquın Abellán, Carlos J. í, and Javier G. Castellano. “AdaptativeCC4.5: Credal C4.5 with a rough class noise estimator”. In: *Expert Systems with Applications* 92.Supplement C (2018), pp. 363–379. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2017.09.057.

[19] Joaquın Abellán, Griselda López, and Juan de Oña. “Analysis of traffic accident severity using Decision Rules via Decision Trees”. In: *Expert Systems with Applications* 40.15 (2013), pp. 6047–6054. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2013.05.027.

[20] Joaquın Abellán and Carlos J. Mantas. “Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring”. In: *Expert Systems with Applications* 41.8 (2014), pp. 3825–3830. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2013.12.003.

[21] Joaquın Abellán and Andrés Masegosa. “Requirements for total uncertainty measures in Dempster-Shafer theory of evidence”. In: *International Journal of General Systems* 37.6 (2008), pp. 733–747.
DOI: 10.1080/03081070802082486.

[22] Joaquın Abellán and Andrés R Masegosa. “An ensemble method using credal decision trees”. In: *European journal of operational research* 205.1 (2010), pp. 218–226.

[23] Joaquın Abellán and Andrés R. Masegosa. “Bagging schemes on the presence of class noise in classification”. In: *Expert Systems with Applications* 39.8 (2012), pp. 6827–6837. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2012.01.013.

[24] S Akila and U Srinivasulu Reddy. “Cost-sensitive Risk Induced Bayesian Inference Bagging (RIBIB) for credit card fraud detection”. In: *Journal of Computational Science* 27 (2018), pp. 247–254. ISSN: 1877-7503.
DOI: 10.1016/j.jocs.2018.06.009.

[25] R. T. Alves, M. R. Delgado, and A. A. Freitas. “Knowledge discovery with Artificial Immune Systems for hierarchical multi-label classification of protein functions”. In: *International Conference on Fuzzy Systems*. 2010, pp. 1–8.
DOI: 10.1109/FUZZY.2010.5584298.

[26] Omer Faruk Arar and Kurşat Ayan. “Software defect prediction using cost-sensitive neural network”. In: *Applied Soft Computing* 33 (2015), pp. 263–277. ISSN: 1568-4946.
DOI: 10.1016/j.asoc.2015.04.045.

[27] T. Augustin and F.P.A. Coolen. “Nonparametric predictive inference and interval probability”. In: *Journal of Statistical Planning and Inference* 124.2 (2004), pp. 251–272. ISSN: 0378-3758.
DOI: 10.1016/j.jspi.2003.07.003.

[28] Bart Baesens, Michael Egmont-Petersen, Robert Castelo, and Jan Vanthienen. “Learning Bayesian network classifiers for credit scoring using Markov Chain Monte Carlo search”. In: *16th International Conference on Pattern Recognition*. Vol. 3. IEEE, 2002, pp. 49–52.
DOI: 10.1109/ICPR.2002.1047792.

[29] Zafer Barutcuoglu, Robert E. Schapire, and Olga G. Troyanskaya. “Hierarchical multi-label prediction of gene function”. In: *Bioinformatics* 22.7 (2006), pp. 830–836.
DOI: 10.1093/bioinformatics/btk048.

[30] Otman Basir and Xiaohong Yuan. “Engine fault diagnosis based on multi-sensor information fusion using Dempster–Shafer evidence theory”. In: *Information Fusion* 8.4 (2007), pp. 379–386. ISSN: 1566-2535.
DOI: 10.1016/j.inffus.2005.07.003.

[31] Thomas Bayes. “LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S”. In: *Philosophical transactions of the Royal Society of London* 53 (1763), pp. 370–418.
DOI: 10.1098/rstl.1763.0053.

[32] Jean-Marc Bernard. “An introduction to the imprecise Dirichlet model for multinomial data”. In: *International Journal of Approximate Reasoning* 39.2 (2005). Imprecise Probabilities and Their Applications, pp. 123–150. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2004.10.002.

[33] José M Bernardo and Adrian FM Smith. *Bayesian theory*. Vol. 405. John Wiley & Sons, 2009.

[34] Malcolm Beynon, Bruce Curry, and Peter Morgan. “The Dempster–Shafer theory of evidence: an alternative approach to multicriteria decision modelling”. In: *Omega* 28.1 (2000), pp. 37–50. ISSN: 0305-0483.
DOI: https://doi.org/10.1016/S0305-0483(99)00033-X.

[35] Soufiane Boufous, Rebecca Ivers, Teresa Senserrick, and Mark Stevenson. “Attempts at the Practical On-Road Driving Test and the Hazard Perception Test and the Risk of Traffic Crashes in Young Drivers”. In: *Traffic Injury Prevention* 12.5 (2011), pp. 475–482. DOI: 10.1080/15389588.2011.591856.

[36] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. “Learning multi-label scene classification”. In: *Pattern Recognition* 37.9 (2004), pp. 1757–1771. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2004.03.009.

[37] Leo Breiman. “Bagging Predictors”. In: *Machine Learning* 24.2 (1996), pp. 123–140. ISSN: 1573-0565. DOI: 10.1023/A:1018054314350.

[38] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324.

[39] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. *Classification and regression trees*. London: Chapman & Hall/CRC, 1984.

[40] Andrey Bronevich and George J. Klir. “Measures of uncertainty for imprecise probabilities: An axiomatic approach”. In: *International Journal of Approximate Reasoning* 51.4 (2010), pp. 365–390. ISSN: 0888-613X. DOI: 10.1016/j.ijar.2009.11.003.

[41] D. M. Buede and P. Girardi. “A target identification comparison of Bayesian and Dempster-Shafer multisensor fusion”. In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 27.5 (1997), pp. 569–577. ISSN: 1558-2426. DOI: 10.1109/3468.618256.

[42] Wray Buntine. “Theory Refinement on Bayesian Networks”. In: *Uncertainty Proceedings 1991*. San Francisco (CA): Morgan Kaufmann, 1991, pp. 52–60. ISBN: 978-1-55860-203-8. DOI: 10.1016/B978-1-55860-203-8.50010-3.

[43] Luis M de Campos and Juan F Huete. *Independence concepts in upper and lower probabilities*. North-Holland, Amsterdam, 1993.

[44] Longbing Cao. “Data Science: Challenges and Directions”. In: *Commun. ACM* 60.8 (2017), pp. 59–68. ISSN: 0001-0782. DOI: 10.1145/3015456.

[45] Yonatan Carlos Carranza Alarcón and Sébastien Destercke. "Distributionally Robust, Skeptical Binary Inferences in Multi-label Problems". In: *Proceedings of the Twelveth International Symposium on Imprecise Probability: Theories and Applications*. Ed. by Andrés Cano, Jasper De Bock, Enrique Miranda, and Serafín Moral. Vol. 147. Proceedings of Machine Learning Research. PMLR, 2021, pp. 51–60.

[46] Yonatan Carlos Carranza Alarcón and Sébastien Destercke. "Multi-label Chaining with Imprecise Probabilities". In: *Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 16th European Conference, ECSQARU 2021, Prague, Czech Republic, September 21-24, 2021, Proceedings*. Ed. by Jirina Vejnarová and Nic Wilson. Vol. 12897. Lecture Notes in Computer Science. Springer, 2021, pp. 413–426.

[47] Bojan Cestnik. "Estimating Probabilities: A Crucial Task in Machine Learning". In: *Proceedings of the 9th European Conference on Artificial Intelligence (ECAI'90)*. London Pitman Publishing, 1990, pp. 147–149.

[48] Bojan Cestnik and Ivan Bratko. "On estimating probabilities in tree pruning". In: *European Working Session on Learning (EWSL-91)*. Vol. 482. Springer Berlin Heidelberg, 1991, pp. 138–150. ISBN: 978-3-540-46308-5. DOI: 10.1007/BFb0017010.

[49] Francisco Charte, Antonio J. Rivera, David Charte, Marıa J. del Jesus, and Francisco Herrera. "Tips, guidelines and tools for managing multilabel datasets: The mldr.datasets R package and the Cometa data repository". In: *Neurocomputing* 289 (2018), pp. 68–85. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2018.02.011.

[50] Francisco Charte, Antonio J. Rivera, María J. [del Jesus], and Francisco Herrera. "Addressing imbalance in multilabel classification: Measures and random resampling algorithms". In: *Neurocomputing* 163 (2015), pp. 3–16. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2014.08.091.

[51] Alain Chateauneuf and Jean-Yves Jaffray. "Some characterizations of lower probabilities and other monotone capacities through the use of Möbius inversion". In: *Mathematical Social Sciences* 17.3 (1989), pp. 263–283. ISSN: 0165-4896. DOI: 10.1016/0165-4896(89)90056-5.

[52] Jie Chen, Xizhao Wang, and Junhai Zhai. "Pruning Decision Tree Using Genetic Algorithms". In: *2009 International Conference on Artificial Intelligence and Computational Intelligence*. Vol. 3. 2009, pp. 244–248. DOI: 10.1109/AICI.2009.351.

[53] T. M. Chen and V. Venkataramanan. “Dempster-Shafer theory for intrusion detection in ad hoc networks”. In: *IEEE Internet Computing* 9.6 (2005), pp. 35–41. ISSN: 1941-0131.
DOI: 10.1109/MIC.2005.123.

[54] Weiwei Cheng and Eyke Hüllermeier. “Combining instance-based learning and logistic regression for multilabel classification”. In: *Machine Learning* 76.2-3 (2009), pp. 211–225.
DOI: 10.1007/s10994-009-5127-5.

[55] G Choquet. “Théorie des capacités”. In: *Annales de l'Institut Fourier*. Vol. 5, pp. 1953–1954.

[56] Amanda Clare and Ross D. King. “Knowledge Discovery in Multi-label Phenotype Data”. In: *Principles of Data Mining and Knowledge Discovery*. Ed. by Luc De Raedt and Arno Siebes. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 42–53. ISBN: 978-3-540-44794-8.

[57] Barry R. Cobb and Prakash P. Shenoy. “On the plausibility transformation method for translating belief function models to probability models”. In: *International Journal of Approximate Reasoning* 41.3 (2006), pp. 314–330. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2005.06.008.

[58] F. P. A. Coolen. “Learning from multinomial data: a nonparametric predictive alternative to the Imprecise Dirichlet Model”. In: *ISIPTA'05: Proceedings of the Fourth International Symposium on Imprecise Probabilities and their Applications, Fabio G. Cozman, Robert Nau and Teddy Seidenfeld (Editors). (Published by SIPTA*. 2005, pp. 125–134.

[59] F.P.A. Coolen and T. Augustin. “A nonparametric predictive alternative to the Imprecise Dirichlet Model: The case of a known number of categories”. In: *International Journal of Approximate Reasoning* 50.2 (2009), pp. 217–230. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2008.03.011.

[60] T. Coolen-Maturi and F. P. A. Coolen. “Non-parametric predictive inference for the validation of credit rating systems”. In: *Journal of the Royal Statistical Society* 182.4 (2019), pp. 1189–1204.
DOI: 10.1111/rssa.12416.

[61] Gregory F. Cooper and Edward Herskovits. “A Bayesian method for the induction of probabilistic networks from data”. In: *Machine Learning* 9.4 (1992), pp. 309–347. ISSN: 1573-0565.
DOI: 10.1007/BF00994110.

[62] Giorgio Corani and Marco Zaffalon. "Learning Reliable Classifiers from Small or Incomplete Data Sets: the Naive Credal Classifier 2". In: *Journal of Machine Learing Research* 9 (2008), pp. 581–621.

[63] Inés Couso, Serafín Moral, and Peter Walley. "A survey of concepts of independence for imprecise probabilities". In: *Risk, Decision and Policy* 5.2 (2000), pp. 165–181.
DOI: 10.1017/S1357530900000156.

[64] T. Cover and P. Hart. "Nearest neighbor pattern classification". In: *IEEE Transactions on Information Theory* 13.1 (1967), pp. 21–27. ISSN: 0018-9448.
DOI: 10.1109/TIT.1967.1053964.

[65] Fabio G. Cozman. "Credal networks". In: *Artificial Intelligence* 120.2 (2000), pp. 199–233. ISSN: 0004-3702.
DOI: 10.1016/S0004-3702(00)00029-1.

[66] Fabio G. Cozman. "Credal networks". In: *Artificial Intelligence* 120.2 (2000), pp. 199–233. ISSN: 0004-3702.
DOI: 10.1016/S0004-3702(00)00029-1.

[67] Fabio G. Cozman and Peter Walley. "Graphoid properties of epistemic irrelevance and independence". In: *Annals of Mathematics and Artificial Intelligence* (2005), pp. 173–195. ISSN: 1012-2443.
DOI: 10.1007/s10472-005-9004-z.

[68] Fabio Gagliardi Cozman. "Graphical models for imprecise probabilities". In: *International Journal of Approximate Reasoning* 39.2 (2005), pp. 167–184. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2004.10.003.

[69] Huizi Cui, Qing Liu, Jianfeng Zhang, and Bingyi Kang. "An Improved Deng Entropy and Its Application in Pattern Recognition". In: *IEEE Access* 7 (2019), pp. 18284–18292.
DOI: 10.1109/ACCESS.2019.2896286.

[70] Luis M. De Campos, Juan F. Huete, and Serafin Moral. "Probability intervals: a tool for uncertain reasoning". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 02.02 (1994), pp. 167–196.
DOI: 10.1142/S0218488594000146.

[71] Luis M. De Campos and Serafin Moral. "Independence Concepts for Convex Sets of Probabilities". In: *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 1995, pp. 108–115. ISBN: 1558603859.

[72] Luis Miguel de Campos Ibañez and Manuel Jorge Bolaños Carmona. "Representation of fuzzy measures through probabilities". In: *Fuzzy Sets and Systems* 31.1 (1989), pp. 23–36. ISSN: 0165-0114. DOI: 10.1016/0165-0114(89)90064-X.

[73] Krzysztof Dembczyński, Weiwei Cheng, and Eyke Hüllermeier. "Bayes Optimal Multilabel Classification via Probabilistic Classifier Chains". In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. Omnipress, 2010, pp. 279–286. ISBN: 9781605589077.

[74] A. P. Dempster. "Upper and Lower Probabilities Induced by a Multivalued Mapping". In: *The Annals of Mathematical Statistics* 38.2 (1967), pp. 325–339. DOI: 10.1214/aoms/1177698950.

[75] Janez Demšar. "Statistical Comparisons of Classifiers over Multiple Data Sets". In: *Journal of Machine Learning Research* 7 (2006), pp. 1–30. ISSN: 1532-4435.

[76] Xinyang Deng. "Analyzing the monotonicity of belief interval based uncertainty measures in belief function theory". In: *International Journal of Intelligent Systems* 33.9 (2018), pp. 1869–1879. DOI: 10.1002/int.21999.

[77] Yong Deng. "Deng entropy". In: *Chaos, Solitons and Fractals* 91 (2016), pp. 549–553. ISSN: 0960-0779. DOI: 110.1016/j.chaos.2016.07.014.

[78] T. Denoeux. "A k-nearest neighbor classification rule based on Dempster-Shafer theory". In: *IEEE Transactions on Systems, Man, and Cybernetics* 25.5 (1995), pp. 804–813. ISSN: 2168-2909. DOI: 10.1109/21.376493.

[79] S. Destercke, D. Dubois, and E. Chojnacki. "Unifying practical uncertainty representations – I: Generalized p-boxes". In: *International Journal of Approximate Reasoning* 49.3 (2008), pp. 649–663. ISSN: 0888-613X. DOI: 10.1016/j.ijar.2008.07.003.

[80] Thomas G. Dietterich. "Ensemble Methods in Machine Learning". In: *Proceedings of the First International Workshop on Multiple Classifier Systems*. MCS '00. London, UK, UK: Springer-Verlag, 2000, pp. 1–15. ISBN: 3-540-67704-6.

[81] Pedro Domingos. "MetaCost: A General Method for Making Classifiers Cost-Sensitive". In: *In Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*. ACM Press, 1999, pp. 155–164.

[82] Pedro Domingos and Michael Pazzani. “On the Optimality of the Simple Bayesian Classifier under Zero-One Loss”. In: *Machine Learning* 29.2 (1997), pp. 103–130. ISSN: 1573-0565.
DOI: 10.1023/A:1007413511361.

[83] Didier Dubois and Henri Prade. “A note on measures of specificity for fuzzy sets”. In: *International Journal of General Systems* 10.4 (1985), pp. 279–283.
DOI: 10.1080/03081078508934893.

[84] Didier Dubois and Henri Prade. “Properties of measures of information in evidence and possibility theories”. In: *Fuzzy Sets and Systems* 24.2 (1987), pp. 161–182. ISSN: 0165-0114.
DOI: 10.1016/0165-0114(87)90088-1.

[85] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. New York: John Wiley and Sons, 1973.

[86] Andre Elisseeff and Jason Weston. “A kernel method for multi-labelled classification”. In: *In Advances in Neural Information Processing Systems 14*. Vol. 14. 2001, pp. 681–687.

[87] L. Enrique Sucar, Concha Bielza, Eduardo F. Morales, Pablo Hernandez-Leal, Julio H. Zaragoza, and Pedro Larra naga. “Multi-label classification with Bayesian network-based chain classifiers”. In: *Pattern Recognition Letters* 41 (2014), pp. 14–22. ISSN: 0167-8655.
DOI: 10.1016/j.patrec.2013.11.007.

[88] U.M. Fayyad and K.B. Irani. “Multi-valued Interval Discretization of Continuous-valued Attributes for classification Learning”. In: *Proceeding of the 13th International joint Conference on Artificial Inteligence*. Morgan Kaufmann, 1993, pp. 1022–1027.

[89] Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C. Prati, Bartosz Krawczyk, and Francisco Herrera. “Cost-Sensitive Learning”. In: *Learning from Imbalanced Data Sets*. Cham: Springer International Publishing, 2018, pp. 63–78. ISBN: 978-3-319-98074-4.
DOI: 10.1007/978-3-319-98074-4_4.

[90] S. Ferson and W.T. Tucker. “Probability boxes as info-gap models”. In: *NAFIPS 2008 - 2008 Annual Meeting of the North American Fuzzy Information Processing Society*. 2008, pp. 1–6.
DOI: 10.1109/NAFIPS.2008.4531314.

[91] Yoav Freund and Robert E. Schapire. “Experiments with a New Boosting Algorithm”. In: *Proceedings of the Thirteenth International Conference on Machine Learning (ICML 1996)*. Ed. by Lorenza Saitta. Morgan Kaufmann, 1996, pp. 148–156. ISBN: 1-55860-419-7.

[92] Milton Friedman. “A Comparison of Alternative Tests of Significance for the Problem of m Rankings”. In: *The Annals of Mathematical Statistics* 11.1 (1940), pp. 86–92.
DOI: 10.1214/aoms/1177731944.

[93] Nir Friedman, Dan Geiger, and Moises Goldszmidt. “Bayesian Network Classifiers”. In: *Machine Learning* 29.2 (1997), pp. 131–163. ISSN: 1573-0565.
DOI: 10.1023/A:1007465528199.

[94] Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencorthogonal a, and Klaus Brinker. “Multilabel classification via calibrated label ranking”. In: *Machine Learning* 73 (2008), pp. 133–153.
DOI: 10.1007/s10994-008-5064-8.

[95] V. García, J.S. Sánchez, and R.A. Mollineda. “On the effectiveness of preprocessing methods when dealing with different levels of class imbalance”. In: *Knowledge-Based Systems* 25.1 (2012), pp. 13–21. ISSN: 0950-7051.
DOI: 10.1016/j.knosys.2011.06.013.

[96] Nadia Ghamrawi and Andrew McCallum. “Collective Multi-Label Classification”. In: *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*. CIKM ’05. New York, NY, USA: Association for Computing Machinery, 2005, pp. 195–200. ISBN: 1595931406.
DOI: 10.1145/1099554.1099591.

[97] Eva Gibaja and Sebastián Ventura. “Multi-label learning: a review of the state of the art and ongoing research”. In: *WIREs Data Mining and Knowledge Discovery* 4.6 (2014), pp. 411–444.
DOI: 10.1002/widm.1139.

[98] Kenneth R Ginsburg, Flaura K Winston, Teresa M Senserrick, Felipe Garcıa-España, Sara Kinsman, D Alex Quistberg, James G Ross, and Michael R Elliott. “National young-driver survey: teen perspective and experience with factors that affect driving safety”. In: *Pediatrics* 121.5 (2008), e1391–e1403.
DOI: 10.1542/peds.2007-2595.

[99] Shantanu Godbole and Sunita Sarawagi. "Discriminative Methods for Multi-labeled Classification". In: *Advances in Knowledge Discovery and Data Mining*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 22–30. ISBN: 978-3-540-24775-3.

[100] E. C. Gonçalves, A. Plastino, and A. A. Freitas. "A Genetic Algorithm for Optimizing the Label Ordering in Multi-label Classifier Chains". In: *2013 IEEE 25th International Conference on Tools with Artificial Intelligence*. 2013, pp. 469–476.
DOI: 10.1109/ICTAI.2013.76.

[101] Eduardo C. Gonçalves, Alexandre Plastino, and Alex A. Freitas. "Simpler is Better: A Novel Genetic Algorithm to Induce Compact Multi-Label Chain Classifiers". In: New York, NY, USA: Association for Computing Machinery, 2015, pp. 559–566. ISBN: 9781450334723.
DOI: 10.1145/2739480.2754650.

[102] Irving John Good. *The Estimation of Probabilities. An essay on modern Bayesian Methods*. Vol. 30. Volume 30 of MIT Research monograph. The M. I. T. Press Cambridge, 1965.
DOI: 10.1002/bimj.19680100118.

[103] Isidore Jacob Good. "Probability and the Weighing of Evidence". In: *British Journal of Social Medicine* 4.3 (1950), pp. 170–171.

[104] Michel Grabisch. "The interaction and Möbius representations of fuzzy measures on finite spaces, k-additive measures: a survey". In: *Fuzzy Measures and Integrals — Theory and Applications*. Physica Verlag, 200, pp. 70–93.

[105] Peijun Guo and Hideo Tanaka. "Decision making with interval probabilities". In: *European Journal of Operational Research* 203.2 (2010), pp. 444–454. ISSN: 0377-2217.
DOI: 10.1016/j.ejor.2009.07.020.

[106] M. A. Hall. "Correlation-based Feature Subset Selection for Machine Learning". PhD thesis. Hamilton, New Zealand: University of Waikato, 1998.

[107] David J. Hand. *Construction and Assessment of Classification Rules*. John Wiley and Sons, New York, 1997. ISBN: 0-471-96583-9.

[108] David J. Hand and Veronica Vinciotti. "Choosing k for two-class nearest neighbour classifiers with unbalanced classes". In: *Pattern Recognition Letters* 24.9 (2003), pp. 1555–1562. ISSN: 0167-8655.
DOI: 0.1016/S0167-8655(02)00394-X.

[109] David Harmanec and George J. Klir. "Measuring total uncertainty in Dempster-Shafer Theory: A novel aaproach". In: *International Journal of General Systems* 22.4 (1994), pp. 405–419. DOI: 10.1080/03081079408935225.

[110] R. V. L. Hartley. "Transmission of Information1". In: *Bell System Technical Journal* 7.3 (1928), pp. 535–563. DOI: 10.1002/j.1538-7305.1928.tb01236.x.

[111] Ting He, Frank P. A. Coolen, and Tahani Coolen-Maturi. "Nonparametric predictive inference for European option pricing based on the binomial tree model". In: *Journal of the Operational Research Society* 70.10 (2019), pp. 1692–1708. DOI: 10.1080/01605682.2018.1495997.

[112] David Heckerman, Dan Geiger, and David M. Chickering. "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data". In: *Machine Learning* 20.3 (1995), pp. 197–243. ISSN: 1573-0565. DOI: 10.1023/A:1022623210503.

[113] Bruce M Hill. "De Finetti's Theorem, Induction, and A (n) or Bayesian nonparametric predictive inference (with discussion)". In: *Bayesian statistics* 3 (1988), pp. 211–241.

[114] T.K. Ho. "The random subspace method for constructing decision forests". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.8 (1998), pp. 832–844. DOI: 10.1109/34.709601.

[115] Sture Holm. "A Simple Sequentially Rejective Multiple Test Procedure". In: *Scandinavian Journal of Statistics* 6.2 (1979), pp. 65–70. ISSN: 03036898, 14679469.

[116] Jun Huang, Guorong Li, Shuhui Wang, Zhe Xue, and Qingming Huang. "Multi-label classification by exploiting local positive and negative pairwise label correlation". In: *Neurocomputing* 257 (2017), pp. 164–174. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2016.12.073.

[117] Peter J Huber. *Robust statistics*. Vol. 523. John Wiley & Sons, 1981.

[118] V.-N. Huynh and Y. Nakamori. "Notes on reducing algorithm complexity for computing an aggregate uncertainty measure". In: *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* 40.1 (2010), pp. 205–209. DOI: 10.1109/TSMCA.2009.2030962.

[119] M. Ioannou, G. Sakkas, G. Tsoumakas, and I. Vlahavas. "Obtaining Bipartitions from Score Vectors for Multi-Label Classification". In: *2010 22nd IEEE International Conference on Tools with Artificial Intelligence*. Vol. 1. 2010, pp. 409–416.
DOI: 10.1109/ICTAI.2010.65.

[120] H. H S Ip and J. M C Ng. "Human face recognition using Dempster-Shafer theory". In: *Proceedings - International Conference on Image Processing, ICIP* 2 (1994), pp. 292–295. ISSN: 1522-4880.
DOI: 10.1109/ICIP.1994.413578.

[121] Serafín Moral-García Javier G. Castellano, María D. Benítez Carlos J. Mantas, and Joaquín Abellán. "A Decision Support Tool for Credit Domains: Bayesian Network with a Variable Selector Based on Imprecise Probabilities". In: *International Journal of Fuzzy Systems* 23 (2021), pp. 2004–2020.
DOI: 10.1007/s40815-021-01079-w.

[122] Radim Jirousek and Prakash P. Shenoy. "A new definition of entropy of belief functions in the Dempster–Shafer theory". In: *International Journal of Approximate Reasoning* 92 (2018), pp. 49–65. ISSN: 0888-613X.
DOI: doi.org/10.1016/j.ijar.2017.10.010.

[123] A -. Jousselme, Chunsheng Liu, D. Grenier, and E. Bosse. "Measuring ambiguity in the evidence theory". In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 36.5 (2006), pp. 890–903. ISSN: 1558-2426.
DOI: 10.1109/TSMCA.2005.853483.

[124] Xie Jun, Yu Lu, Zhu Lei, and Duan Guolun. "Conditional entropy based classifier chains for multi-label classification". In: *Neurocomputing* 335 (2019), pp. 185–194. ISSN: 0925-2312.
DOI: 10.1016/j.neucom.2019.01.039.

[125] Bingyi Kang and Yong Deng. "The Maximum Deng Entropy". In: *IEEE Access* 7 (2019), pp. 120758–120765.
DOI: 10.1109/ACCESS.2019.2937679.

[126] Kenji Kira and Larry A. Rendell. "The Feature Selection Problem: Traditional Methods and a New Algorithm". In: *Proceedings of the Tenth National Conference on Artificial Intelligence*. AAAI Press, 1992, pp. 129–134. ISBN: 0262510634.

[127] G. Klir and M. Wierman. *Uncertainty-Based Information: Elements of Generalized Information Theory*. Studies in Fuzziness and Soft Computing. Physica-Verlag HD, 1999.

[128] George J. Klir. *Uncertainty and Information: Foundations of Generalized Information Theory*. John Wiley and Sons, Inc., 2005. ISBN: 9780471755579. DOI: 10.1002/0471755575.

[129] George J. Klir and Harold W. Lewis. "Remarks on "Measuring Ambiguity in the Evidence Theory"". In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 38.4 (2008), pp. 995–999. DOI: 10.1109/TSMCA.2008.923066.

[130] George J. Klir and Richard M. Smith. "On Measuring Uncertainty and Uncertainty-Based Information: Recent Developments". In: *Annals of Mathematics and Artificial Intelligence* 32.1 (2001), pp. 5–33. DOI: 10.1023/A:1016784627561.

[131] Dongo Rémi Kouabenan. "Occupation, driving experience, and risk and accident perception". In: *Journal of Risk Research* 5.1 (2002), pp. 49–68.

[132] S. Kullback. "Probability Densities with Given Marginals". In: *The Annals of Mathematical Statistics* 39.4 (1968), pp. 1236–1243. DOI: 10.1214/aoms/1177698249.

[133] Pat Langley, Wayne Iba, and Kevin Thompson. "An analysis of Bayesian classifiers". In: *Proceedings of the 10th national conference on Artificial intelligence (AAAI 92)*. MIT Press, 1992, pp. 223–228.

[134] Jaedong Lee, Heera Kim, Noo-ri Kim, and Jee-Hyong Lee. "An approach for multi-label classification by directed acyclic graph with label correlation maximization". In: *Information Sciences* 351 (2016), pp. 101–114. ISSN: 0020-0255. DOI: 10.1016/j.ins.2016.02.037.

[135] John F. Lemmer and Henry E. Kyburg. "Conditions for the Existence of Belief Functions Corresponding to Intervals of Belief". In: *Proceedings of the Ninth National Conference on Artificial Intelligence - Volume 1*. AAAI Press, 1991, pp. 488–493. ISBN: 0262510596.

[136] Chee Kian Leong. "Credit Risk Scoring with Bayesian Network Models". In: *Computational Economics* 47.3 (2016), pp. 423–446. ISSN: 1572-9974. DOI: 10.1007/s10614-015-9505-8.

[137] Isaac Levi. *The enterprise of knowledge: An essay on knowledge, credal probability, and chance*. MIT press, 1983.

[138] Feng Li, Duoqian Miao, and Witold Pedrycz. “Granular multi-label feature selection based on mutual information”. In: *Pattern Recognition* 67 (2017), pp. 410–423. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2017.02.025.

[139] M. Lichman. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[140] C.X. Ling, V.S. Sheng, and Q. Yang. “Test strategies for cost-sensitive decision trees”. In: *IEEE Transactions on Knowledge and Data Engineering* 18.8 (2006), pp. 1055–1067. DOI: 10.1109/TKDE.2006.131.

[141] Charles X. Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. “Decision Trees with Minimal Costs”. In: *Proceedings of the Twenty-First International Conference on Machine Learning*. ICML ’04. New York, NY, USA: Association for Computing Machinery, 2004, p. 69. ISBN: 1581138385. DOI: 10.1145/1015330.1015369.

[142] Mingxia Liu, Linsong Miao, and Daoqiang Zhang. “Two-Stage Cost-Sensitive Learning for Software Defect Prediction”. In: *IEEE Transactions on Reliability* 63.2 (2014), pp. 676–686. DOI: 10.1109/TR.2014.2316951.

[143] R Duncan Luce and Howard Raiffa. *Games and decisions: Introduction and critical survey*. Courier Corporation, 1989.

[144] Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj, and Sašo Džeroski. “An extensive experimental comparison of methods for multi-label learning”. In: *Pattern Recognition* 45.9 (2012), pp. 3084–3104. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2012.03.004.

[145] Y. Maeda, H. T. Nguyen, and H. Ichihashi. “Maximum entropy algorithms for uncertainty measures”. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 01.01 (1993), pp. 69–93. DOI: 10.1142/S021848859300005X.

[146] Sam Maes, Karl Tuyls, Bram Vanschoenwinkel, and Bernard Manderick. “Credit card fraud detection using Bayesian and neural networks”. In: *Proceedings of the 1st international NAISO congress on neuro fuzzy technologies*. 2002, pp. 261–270.

[147] O. Maimon and L. Rokach. *Classification trees. Data Mining and Knowledge Discovery Handbook*. NJ, USA: Springer-Verlag New York, 2010. DOI: 10.1007/978-0-387-09823-4.

[148] Carlos J. Mantas and Joaquín Abellán. "Analysis and extension of decision trees based on imprecise probabilities: Application on noisy data". In: *Expert Systems with Applications* 41.5 (2014), pp. 2514–2525. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2013.09.050.

[149] Carlos J. Mantas and Joaquín Abellán. "Credal-C4.5: Decision tree based on imprecise probabilities to classify noisy data". In: *Expert Systems with Applications* 41.10 (2014), pp. 4625–4637. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2014.01.017.

[150] Carlos J. Mantas, Joaquín Abellán, and Javier G. Castellano. "Analysis of Credal-C4.5 for classification in noisy domains". In: *Expert Systems with Applications* 61 (2016), pp. 314–326. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2016.05.035.

[151] AI Marqués, Vicente Garcıa, and Javier Salvador Sánchez. "Exploring the behaviour of base classifiers in credit scoring ensembles". In: *Expert Systems with Applications* 39.11 (2012), pp. 10244–10250. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2012.02.092.

[152] Andrew McCallum. "Multi-label text classification with a mixture model trained by EM". In: *AAAI'99 Workshop on Text Learning*. 1999, pp. 1–7.

[153] Prem Melville and Raymond J. Mooney. "Creating diversity in ensembles using artificial data". In: *Information Fusion* 6.1 (2005), pp. 99–111. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2004.04.001.

[154] Aaron Meyerowitz, Fred Richman, and Elbert Walker. "Calculating maximum-entropy probability densities for belief functions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 02.04 (1994), pp. 377–389. DOI: 10.1142/S0218488594000316.

[155] Enrique Miranda. "Updating coherent previsions on finite spaces". In: *Fuzzy Sets and Systems* 160.9 (2009). Theme: Topology and non-Additive Measures, pp. 1286–1307. ISSN: 0165-0114. DOI: 10.1016/j.fss.2008.10.005.

[156] Enrique Miranda and Sébastien Destercke. "Extreme points of the credal sets generated by comparative probabilities". In: *Journal of Mathematical Psychology* 64-65 (2015), pp. 44–57. ISSN: 0022-2496. DOI: 10.1016/j.jmp.2014.11.004.

[157] Enrique Miranda and Ignacio Montes. "Coherent updating of non-additive measures". In: *International Journal of Approximate Reasoning* 56 (2015), pp. 159–177. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2014.05.003.

[158] Enrique Miranda, Marco Zaffalon, and Gert Cooman. "Conglomerable natural extension". In: *International Journal of Approximate Reasoning* 53.8 (2012), pp. 1200–1227. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2012.06.015.

[159] W. Nor Haizan W. Mohamed, Mohd Najib Mohd Salleh, and Abdul Halim Omar. "A comparative study of Reduced Error Pruning method in decision tree algorithms". In: *2012 IEEE International Conference on Control System, Computing and Engineering*. 2012, pp. 392–397.
DOI: 10.1109/ICCSCE.2012.6487177.

[160] Alfonso Montella, Massimo Aria, Antonio D'Ambrosio, and Filomena Mauriello. "Analysis of powered two-wheeler crashes in Italy by classification trees and rules discovery". In: *Accident Analysis & Prevention* 49.Supplement C (2012), pp. 58–72. ISSN: 0001-4575.
DOI: 10.1016/j.aap.2011.04.025.

[161] Ignacio Montes and Sebastien Destercke. "On extreme points of p-boxes and belief functions". In: *Annals of Mathematics and Artificial Intelligence* 81 (2017), pp. 405–428.
DOI: 10.1007/s10472-017-9562-x.

[162] Ignacio Montes, Enrique Miranda, and Susana Montes. "Decision making with imprecise probabilities and utilities by means of statistical preference and stochastic dominance". In: *European Journal of Operational Research* 234.1 (2014), pp. 209–220. ISSN: 0377-2217.
DOI: 10.1016/j.ejor.2013.09.013.

[163] Ignacio Montes, Enrique Miranda, and Susana Montes. "Stochastic dominance with imprecise information". In: *Computational Statistics & Data Analysis* 71 (2014), pp. 868–886. ISSN: 0167-9473.
DOI: 10.1016/j.csda.2012.07.030.

[164] Serafín Moral-García, Javier G. Castellano, Carlos J. Mantas, Alfonso Montella, and Joaquín Abellán. "Decision Tree Ensemble Method for Analyzing Traffic Accidents of Novice Drivers in Urban Areas". In: *Entropy* 21.4 (2019). ISSN: 1099-4300.
DOI: 10.3390/e21040360.

[165] R. R. Murphy. “Dempster-Shafer theory for sensor fusion in autonomous mobile robots”. In: *IEEE Transactions on Robotics and Automation* 14.2 (1998), pp. 197–206. ISSN: 2374-958X.
DOI: 10.1109/70.681240.

[166] Sanaz Nami and Mehdi Shajari. “Cost-sensitive payment card fraud detection based on dynamic random forest and k-nearest neighbors”. In: *Expert Systems with Applications* 110 (2018), pp. 381–392. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2018.06.011.

[167] Peter Nemenyi. “Distribution-free multiple comparisons”. Doctoral dissertation. New Jersey, USA: Princeton University, 1963.

[168] Vu-Linh Nguyen and Eyke Hullermeier. “Reliable Multilabel Classification: Prediction with Partial Abstention”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.04 (2020), pp. 5264–5271.
DOI: 10.1609/aaai.v34i04.5972.

[169] Juan de Oña, Griselda López, and Joaquın Abellán. “Extracting decision rules from police accident reports through decision trees”. In: *Accident Analysis & Prevention* 50 (2013), pp. 1151–1160. ISSN: 0001-4575.
DOI: 10.1016/j.aap.2012.09.006.

[170] Qian Pan, Deyun Zhou, Yongchuan Tang, Xiaoyang Li, and Jichuan Huang. “A Novel Belief Entropy for Measuring Uncertainty in Dempster-Shafer Evidence Theory Framework Based on Plausibility Transformation and Weighted Hartley Entropy”. In: *Entropy* 21.2 (2019), p. 163. ISSN: 1099-4300.
DOI: 10.3390/e21020163.

[171] Yoon-Joo Park, Se-Hak Chun, and Byung-Chun Kim. “Cost-sensitive case-based reasoning using a genetic algorithm: Application to medical diagnosis”. In: *Artificial Intelligence in Medicine* 51.2 (2011), pp. 133–145. ISSN: 0933-3657.
DOI: 10.1016/j.artmed.2010.12.001.

[172] Jude Pearl. *Probabilistic reasoning in intelligent systems*. Morgan Kaufmann, San Mateo, CA, 1988.

[173] John C. Platt. *Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods*. 1999.

[174] Zhenxing Qin, Alan Tao Wang, Chengqi Zhang, and Shichao Zhang. "Cost-Sensitive Classification with k-Nearest Neighbors". In: *Knowledge Science, Engineering and Management*. Ed. by Mingzheng Wang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 112–131. ISBN: 978-3-642-39787-5.
DOI: 10.1007/978-3-642-39787-5_10.

[175] José Ramón Quevedo, Oscar Luaces, and Antonio Bahamonde. "Multilabel classifiers with a probabilistic thresholding strategy". In: *Pattern Recognition* 45.2 (2012), pp. 876–883. ISSN: 0031-3203.
DOI: 10.1016/j.patcog.2011.08.007.

[176] J. Ross Quinlan. "Induction of Decision Trees". In: *Machine Learning* 1.1 (1986), pp. 81–106. ISSN: 0885-6125.
DOI: 10.1023/A:1022643204877.

[177] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. ISBN: 1-55860-238-0.

[178] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. "Classifier chains for multi-label classification". In: *Machine Learning* 85.3 (2011), p. 333. ISSN: 1573-0565.
DOI: 10.1007/s10994-011-5256-5.

[179] Jesse Read, Peter Reutemann, Bernhard Pfahringer, and Geoff Holmes. "MEKA: A Multi-label/Multi-target Extension to Weka". In: *Journal of Machine Learning Research* 17.21 (2016), pp. 1–5. URL: http://jmlr.org/papers/v17/12-164.html.

[180] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso. "Rotation Forest: A New Classifier Ensemble Method". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.10 (2006), pp. 1619–1630. ISSN: 0162-8828.
DOI: 10.1109/TPAMI.2006.211.

[181] José A Sáez, Julián Luengo, and Francisco Herrera. "Evaluating the classifier behavior with noisy data considering performance and robustness: the Equalized Loss of Accuracy measure". In: *Neurocomputing* 176 (2014), pp. 26–35. ISSN: 0925-2312.
DOI: 10.1016/j.neucom.2014.11.086.

[182] Yusuf Sahin, Serol Bulkan, and Ekrem Duman. "A cost-sensitive decision tree approach for fraud detection". In: *Expert Systems with Applications* 40.15 (2013), pp. 5916–5923. ISSN: 0957-4174.
DOI: 10.1016/j.eswa.2013.05.021.

[183] Raúl Santos-Rodríguez, Darío García-García, and Jeús Cid-Sueiro. "Cost-Sensitive Classification Based on Bregman Divergences for Medical Diagnosis". In: *2009 International Conference on Machine Learning and Applications*. 2009, pp. 551–556.
DOI: 10.1109/ICMLA.2009.82.

[184] Robert E Schapire. "The strength of weak learnability". In: *Machine learning* 5.2 (1990), pp. 197–227.
DOI: 10.1007/BF00116037.

[185] Robert E. Schapire and Yoram Singer. "BoosTexter: A Boosting-based System for Text Categorization". In: *Machine Learning* 39.2 (2000), pp. 135–168. ISSN: 1573-0565.
DOI: 10.1023/A:1007649029923.

[186] Glenn Shafer. *A mathematical theory of evidence*. Ed. by Princeton University Press. Princeton university press Princeton, 1976.

[187] Glenn Shafer. "Belief functions and possibility measures." English (US). In: *Anal of Fuzzy Inf*. CRC Press Inc, 1987, pp. 51–84. ISBN: 0849362962.

[188] A. Shahpari and S. A. Seyedin. "Using Mutual Aggregate Uncertainty Measures in a Threat Assessment Problem Constructed by Dempster–Shafer Network". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45.6 (2015), pp. 877–886. ISSN: 2168-2216.
DOI: 10.1109/TSMC.2014.2378213.

[189] C. E. Shannon. "A Mathematical Theory of Communication". In: *Bell System Technical Journal* 27.3 (1948), pp. 379–423. ISSN: 1538-7305.
DOI: 10.1002/j.1538-7305.1948.tb01338.x.

[190] P. P. Shenoy and G. Shafer. "Readings in Uncertain Reasoning". In: ed. by Glenn Shafer and Judea Pearl. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990. Chap. Axioms for Probability and Belief-function Propagation, pp. 575–610. ISBN: 1-55860-125-2.

[191] Michael J. Siers and Md Zahidul Islam. "Software defect prediction using a cost sensitive decision forest and voting, and a potential solution to the class imbalance problem". In: *Information Systems* 51 (2015), pp. 62–71. ISSN: 0306-4379.
DOI: 10.1016/j.is.2015.02.006.

[192] Richard McKee Smith. *Generalized information theory: resolving some old questions and opening some new ones*. State University of New York at Binghamton, 2000.

[193] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. Vol. 81. Lecture Notes in Statistics. MIT press, 1993. ISBN: 978-1-4612-7650-0.
DOI: 10.1007/978-1-4612-2748-9.

[194] E. Spyromitros, G. Tsoumakas, and Ioannis Vlahavas. "An Empirical Study of Lazy Multilabel Classification Algorithms". In: *Artificial Intelligence: Theories, Models and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 401–406. ISBN: 978-3-540-87881-0.

[195] Patrick Suppes. "The Measurement of Belief". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 36.2 (1974), pp. 160–191. ISSN: 00359246.

[196] Richard Tay. "A random parameters probit model of urban and rural intersection crashes". In: *Accident Analysis & Prevention* 84 (2015), pp. 38–40. ISSN: 0001-4575.
DOI: 10.1016/j.aap.2015.07.013.

[197] B Tessen. *Interval Representation of Uncertainty in Artificial Intelligence*. 1989.

[198] Athanasios Theofilatos, Daniel Graham, and George Yannis. "Factors Affecting Accident Severity Inside and Outside Urban Areas in Greece". In: *Traffic Injury Prevention* 13.5 (2012), pp. 458–467.
DOI: 10.1080/15389588.2012.661110.

[199] Kai Ming Ting. "An instance-weighting method to induce cost-sensitive trees". In: *IEEE Transactions on Knowledge and Data Engineering* 14.3 (2002), pp. 659–665.
DOI: 10.1109/TKDE.2002.1000348.

[200] Matthias C.M. Troffaes. "Decision making under uncertainty using imprecise probabilities". In: *International Journal of Approximate Reasoning* 45.1 (2007), pp. 17–29. ISSN: 0888-613X.
DOI: 10.1016/j.ijar.2006.06.001.

[201] Konstantinos Trohidis, Grigorios Tsoumakas, George Kalliris, and Ioannis P Vlahavas. "Multi-Label Classification of Music into Emotions." In: *ISMIR*. Vol. 8. 2008, pp. 325–330.

[202] Grigorios Tsoumakas, Eleftherios Spyromitros-Xioufis, Jozef Vilcek, and Ioannis Vlahavas. "Mulan: A Java Library for Multi-Label Learning". In: *Journal of Machine Learning Research* 12 (2011), pp. 2411–2414.

[203] Grigorios Tsoumakas and Ioannis Vlahavas. “Random k-Labelsets: An Ensemble Method for Multilabel Classification”. In: *European Conference on Machine Learning*. Springer, 2007, pp. 406–417.
DOI: 10.1007/978-3-540-74958-5_38.

[204] G. Underwood. “Visual attention and the transition from novice to advanced driver”. In: *Ergonomics* 50.8 (2007), pp. 1235–1249.
DOI: 10.1080/00140130701318707.

[205] Geoffrey Underwood, Peter Chapman, Neil Brocklehurst, Jean Underwood, and David Crundall. “Visual attention while driving: sequences of eye fixations made by experienced and novice drivers”. In: *Ergonomics* 46.6 (2003), pp. 629–646.
DOI: 10.1080/0014013031000090116.

[206] Geoffrey Underwood, David Crundall, and Peter Chapman. “Selective searching while driving: the role of experience in hazard detection and general surveillance”. In: *Ergonomics* 45.1 (2002), pp. 1–12.
DOI: 10.1080/00140130110110610.

[207] Frans Voorbraak. “A computationally efficient approximation of Dempster-Shafer theory”. In: *International Journal of Man-Machine Studies* 30.5 (1989), pp. 525–536. ISSN: 0020-7373.
DOI: 10.1016/S0020-7373(89)80032-X.

[208] Peter Walley. *Statistical Reasoning with Imprecise Probabilities*. Vol. 42. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1991.

[209] Peter Walley. “Inferences from multinomial data; learning about a bag of marbles (with discussion)”. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), pp. 3–57. ISSN: 00359246.
DOI: 10.2307/2346164.

[210] Gang Wang, Jian Ma, Lihua Huang, and Kaiquan Xu. “Two credit scoring models based on dual strategy ensemble trees”. In: *Knowledge-Based Systems* 26 (2012), pp. 61–68. ISSN: 0950-7051.
DOI: 10.1016/j.knosys.2011.06.020.

[211] Xiaodan Wang and Yafei Song. “Uncertainty measure in evidence theory with its applications”. In: *Applied Intelligence* 48 (2018), pp. 1672–1688.
DOI: 10.1007/s10489-017-1024-y.

[212] Xinodan Wang and Yafei Song. "Uncertainty measure in evidence theory with its applications". In: *Applied Intelligence* 48 (2017), pp. 1672–1688.
DOI: 10.1007/s10489-017-1024-y.

[213] L Wasserman and JB Kadane. "Bayesian Analysis in Statistics and Econometrics". In: John Wiley, New York, 1996. Chap. 47, pp. 549–555.

[214] W. Weng, D. Wang, C. Chen, J. Wen, and S. Wu. "Label Specific Features-Based Classifier Chains for Multi-Label Classification". In: *IEEE Access* 8 (2020), pp. 51265–51275.
DOI: 10.1109/ACCESS.2020.2980551.

[215] Anna-Stina Wikman, Tapio Nieminen, and Heikki Summala. "Driving experience and time-sharing during in-car tasks on roads of different width". In: *Ergonomics* 41.3 (1998), pp. 358–372.
DOI: 10.1080/001401398187080.

[216] Frank Wilcoxon. "Individual Comparisons by Ranking Methods". In: *Biometrics Bulletin* 1.6 (1945), pp. 80–83. ISSN: 00994987.
DOI: 10.2307/3001968.

[217] Peter M. Williams. "On a New Theory of Epistemic Probability". In: *The British Journal for the Philosophy of Science* 29.4 (1978), pp. 375–387. ISSN: 00070882, 14643537.

[218] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Second. Morgan Kaufmann Series in Data Management Systems. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005. ISBN: 0120884070.

[219] World Health Organization. *Global status report on road safety: time for action*. World Health Organization, 2009.

[220] Deng Xinyang, Xiao Fuyuan, and Deng Yong. "An improved distance-based total uncertainty measure in belief function theory". In: *Applied Intelligence* 46.4 (2017), pp. 898–915.
DOI: 10.1007/s10489-016-0870-3.

[221] Jianhua Xu. "Multi-Label Weighted k-Nearest Neighbor Classifier with Adaptive Weight Estimation". In: *Neural Information Processing*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 79–88. ISBN: 978-3-642-24958-7.

[222] Ronald R. Yager. "Entropy and specificity in a mathematical theory of evidence". In: *International Journal of General Systems* 9.4 (1983), pp. 249–260.
DOI: 10.1080/03081078308960825.

[223] Yi Yang and Deqiang Han. "A new distance-based total uncertainty measure in the theory of belief functions". In: *Knowledge-Based Systems* 94 (2016), pp. 114–123. ISSN: 0950-7051.
DOI: 10.1016/j.knosys.2015.11.014.

[224] Yi Yang, Deqiang Han, and Jean Dezert. "A new non-specificity measure in evidence theory based on belief intervals". In: *Chinese Journal of Aeronautics* 29.3 (2016), pp. 704–713. ISSN: 1000-9361.
DOI: 10.1016/j.cja.2016.03.004.

[225] Xiaomin You and Fulvio Tonon. "Event-Tree Analysis with Imprecise Probabilities". In: *Risk Analysis* 32.2 (2012), pp. 330–344.
DOI: 10.1111/j.1539-6924.2011.01721.x.

[226] Z. Younes, F. Abdallah, and T. Denoeux. "Multi-label classification algorithm derived from K-nearest neighbor rule with label dependencies". In: *2008 16th European Signal Processing Conference*. 2008, pp. 1–5.

[227] Marco Zaffalon. "The naive credal classifier". In: *Journal of Statistical Planning and Inference* 105.1 (2002), pp. 5–21. ISSN: 0378-3758.
DOI: 10.1016/S0378-3758(01)00201-4.

[228] Julio H. Zaragoza, L. Enrique Sucar, Eduardo F. Morales, Concha Bielza, and Pedro Larrañaga. "Bayesian Chain Classifiers for Multidimensional Classification". In: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*. AAAI Press, 2011, pp. 2192–2197. ISBN: 9781577355151.

[229] G.P. Zhang. "Neural networks for classification: a survey". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 30.4 (2000), pp. 451–462.
DOI: 10.1109/5326.897072.

[230] M. L. Zhang and Z. H. Zhou. "A Review on Multi-Label Learning Algorithms". In: *IEEE Transactions on Knowledge and Data Engineering* 26.8 (2014), pp. 1819–1837. ISSN: 1041-4347.
DOI: 10.1109/TKDE.2013.39.

[231] Min-Ling Zhang and Zhi-Hua Zhou. "Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization". In: *IEEE Transactions on Knowledge and Data Engineering* 18.10 (2006), pp. 1338–1351. ISSN: 1041-4347.
DOI: 10.1109/TKDE.2006.162.

[232] Min-Ling Zhang and Zhi-Hua Zhou. "ML-KNN: A lazy learning approach to multi-label learning". In: *Pattern Recognition* 40.7 (2007), pp. 2038–2048. ISSN: 0031-3203.
DOI: 10.1016/j.patcog.2006.12.019.

[233] Ping Zhang, Guixia Liu, and Wanfu Gao. "Distinguishing two types of labels for multi-label feature selection". In: *Pattern Recognition* 95 (2019), pp. 72–82. ISSN: 0031-3203.
DOI: 10.1016/j.patcog.2019.06.004.

[234] Zan Zhang, Hao Wang, Lin Liu, and Jiuyong Li. "Multi-label relational classification via node and label correlation". In: *Neurocomputing* 292 (2018), pp. 72–81. ISSN: 0925-2312.
DOI: 10.1016/j.neucom.2018.02.079.

[235] Yonggang Zhao, Duofa Ji, Xiaodong Yang, Liguo Fei, and Changhai Zhai. "An Improved Belief Entropy to Measure Uncertainty of Basic Probability Assignments Based on Deng Entropy and Belief Interval". In: *Entropy* 21.11 (2019), pp. 1122–1137. ISSN: 1099-4300.
DOI: 10.3390/e21111122.

[236] Deyun Zhou, Yongchuan Tang, and Wen Jiang. "A modified belief entropy in Dempster-Shafer framework". In: *Plos One* 12.5 (2017), pp. 1–17.
DOI: 10.1371/journal.pone.0176832.

[237] Ruonan Zhu, Jiaqi Chen, and Bingyi Kang. "Power Law and Dimension of the Maximum Value for Belief Distribution With the Maximum Deng Entropy". In: *IEEE Access* 8 (2020), pp. 47713–47719.
DOI: 10.1109/ACCESS.2020.2979060.

[238] Yuanhang Zhuang, Zhuoming Xu, and Yan Tang. "A Credit Scoring Model Based on Bayesian Network and Mutual Information". In: *Web Information System and Application Conference (WISA), 2015 12th*. IEEE. 2015, pp. 281–286.
DOI: 10.1109/WISA.2015.31.