UNIVERSIDAD
DE GRANADA

Facultad de Ciencias

Grado en Matemáticas

TRABAJO DE FIN DE GRADO

Desarrollo de modelos de clasificación de imágenes histológicas basados en técnicas de crowdsourcing

Presentado por:
Alberto Díaz-Malaguilla Puntas

Tutor:
Rafael Molina Soriano
CCIA

Miguel López Pérez
CCIA

Curso académico 2020-2021

Desarrollo de modelos de clasificación de imágenes histológicas basados en técnicas de crowdsourcing

Alberto Díaz-Malaguilla Puntas

Alberto Díaz-Malaguilla Puntas *Desarrollo de modelos de clasificación de imágenes histológicas basados en técnicas de crowdsourcing*.

Trabajo de fin de Grado. Curso académico 2020-2021.

Responsable de tutorización

Rafael Molina Soriano
CCIA

Miguel López Pérez
CCIA

Grado en Matemáticas

Facultad de Ciencias

Universidad de Granada

Declaración de originalidad

D./Dña. Alberto Díaz-Malaguilla Puntas

Declaro explícitamente que el trabajo presentado como Trabajo de Fin de Grado (TFG), correspondiente al curso académico 2020-2021, es original, entendida esta, en el sentido de que no ha utilizado para la elaboración del trabajo fuentes sin citarlas debidamente.

En Granada a 16 de junio de 2021

Fdo: Alberto Díaz-Malaguilla Puntas

Índice general

Agradecimientos

En primer lugar, a mis tutores, por su implicación en el trabajo. A Rafael Molina Soriano, por plantearme este reto, por transmitírmelo con entusiasmo, por guiarme y porque su exigencia me ha animado a trabajar duro y dar lo mejor de mí. A Miguel López Pérez, por estar siempre dispuesto a aclarar mis dudas, por su atención a mi trabajo, porque me ha animado en los momentos bajos y me ha hecho el trabajo mucho más fácil.

En segundo lugar, a mi familia, especialmente a mi madre y a Gabri, por su apoyo incondicional y desinteresado. A mi madre, por ayudarme en estos años tan duros a seguir adelante, por todo lo que sacrifica y me da para que alcance mis objetivos, por habérmelo dado todo. A Gabri, por ser el mejor acompañante, por aliviar todas mis cargas, porque sin él los tropiezos hubiesen sido más duros, por hacerme siempre la vida un poco más fácil.

En tercer lugar, a mis amigos y compañeros en la Universidad, Elena y Tarek. Sin su apoyo todo habría sido más amargo y solitario.

A todos ellos se lo agradezco, y les dedico el que creo que será mi último trabajo de Matemáticas.

Summary

The main subject of this work is to develop the theory needed to understand and analyze a classification experiment based on crowdsourcing and apply it to breast cancer histological images.

First, we will introduce basic concepts of variational inference, based on [Bis06], [KF09] and [Pri12]. This is a quite advanced concept for a Mathematics Degree student. Because of that, we will start by introducing Bayesian inference before, based on [Was04] and [Bol07]. While Bayesian inference is explained, we get used to concepts such as prior distribution or posterior distribution, and a new way of estimation that is more general than maximum likelihood estimation, which has been widely explained during Mathematics Degree: maximum a posteriori estimation. With this background, we introduce basic concepts of variational inference: Kullback-Leibler divergence and the evidence lower bound, using concepts of Statistic Inference and Functional Analysis.

Secondly, we will introduce basic concepts of Gaussian processes and apply them to regression, based on [RW06]. To understand them, we will explain Bayesian regression before. As we explain concepts such as basis function, we can comprehend Gaussian processes as models of distributions over functions. This will be useful for introducing regression using Gaussian processes. We will then explain supervised classification tasks using linear models.

Thirdly, we will introduce the color normalization and color transfer problems, based on [RAGS01]. This chapter provides a description of steps that have to be taken before image classification. The algorithm explained will be used on real histological images extracted from the Digital Pathology Asociation repository, [DPA].

To conclude, we will use all what we have learned in the previous chapters to develop a classification model using scalable variational Gaussian processes, specifically on crowdsourcing. We will explain this based on [MARC+20]. We will introduce the crowdsourcing problem as well as classification figures of merits used in machine learning, which are explained on [Gé17]. Finally, we will analyze the breast cancer experiment results from [LPAMÁ+21].

By analyzing this paper the main subject of this work has been achieved, since we have used Statistic Inference, Computational Statistics, programming, and Functional Analysis knowledge from Mathematics Degree, in order to understand variational inference and Gaussian processes theory. Given that, we have applied it to machine learning, especially to crowdsourcing. From the knowledge of a student from a Mathematics degree, we get to approach state-of-the-art data science methods.

Introducción

El objetivo de este trabajo es desarrollar la teoría necesaria para entender y analizar un experimento de aprendizaje automático basado en crowdsourcing aplicado a imágenes histológicas de cáncer de mama.

En primer lugar, desarrollaremos los conceptos básicos de inferencia variacional basándonos esencialmente en [Bis06], [KF09] y [Pri12]. Desde los conocimientos de Inferencia Estadística que tiene un estudiante del Grado en Matemáticas, el salto conceptual a la inferencia variacional puede ser grande. Es por eso que se desarrolla de manera intermedia la inferencia bayesiana, basándonos en [Was04] y [Bol07]. Durante el desarrollo de la inferencia bayesiana, conseguimos familiarizarnos con los conceptos de distribución a priori y distribución a posteriori, y una nueva forma de estimación que generaliza a la estimación máximo verosímil desarrollada durante el Grado en Matemáticas: la estimación máximo a posteriori. Hechas estas consideraciones, conseguimos desarrollar los conceptos básicos de inferencia variacional: la divergencia Kullback-Leibler y la cota inferior de la evidencia, utilizando conceptos de Inferencia Estadística junto a conceptos de Análisis Funcional.

En segundo lugar, desarrollaremos los conceptos básicos de procesos gaussianos aplicados a regresión, basándonos en [RW06]. De nuevo, para desarrollarlos desde el conocimiento de un estudiante del Grado en Matemáticas se desarrolla en primer lugar la regresión desde la inferencia bayesiana. Así, conseguimos familiarizarnos con el concepto de función base, que nos permite entender los procesos gaussianos como modelos de distribución sobre un espacio de funciones. Utilizaremos esto para desarrollar regresión usando procesos gaussianos. Finalmente, se utiliza la teoría de modelos lineales desarrollada para introducir los problemas de clasificación supervisada.

En tercer lugar, haremos una revisión del artículo de normalización de color [RAGS01]. Este capítulo da una idea de cómo es el proceso previo a la tarea de clasificación de imágenes, mediante desarrollos matemáticos sencillos. Aplicaremos el algoritmo planteado en imágenes histológicas extraídas del repositorio de la Digital Pathology Asociation [DPA].

Finalmente, en el cuarto capítulo, aplicamos todo lo desarrollado anteriormente. Desarrollaremos el modelo de clasificación por procesos gaussianos variacionales escalables en el marco del crowdsourcing. Para ello nos basaremos en el artículo [MARC+20]. Introducimos el problema del crowdsourcing, junto con algunos conceptos de aprendizaje automático en el contexto de problemas de clasificación. Finalmente, se analizan los resultados del experimento con imágenes de cáncer de mama que se desarrolla en el artículo [LPAMÁ+21]. Para ello harán falta conceptos de análisis de resultados de aprendizaje automático, que se desarrollarán basándonos en [Gé17].

Con el análisis de este artículo, se alcanza el objetivo del trabajo, utilizando los conceptos aprendidos de Inferencia Estadística, Informática, Análisis Funcional y Estadística Computacional de un estudiante del Grado en Matemáticas para desarrollar la teoría de inferencia variacional y procesos gaussianos. Desarrollada esta teoría, se aplica en aprendizaje automático, en concreto en crowdsourcing, utilizando así los conocimientos del Grado en Matemáticas para aproximarse a temas de investigación de actualidad en el contexto de la ciencia de datos.

1. Inferencia variacional

La Inferencia Estadística se encarga de analizar los datos observados en un experimento aleatorio para extraer conclusiones sobre las propiedades de la distribución de probabilidad subyacente. En el caso de la inferencia paramétrica, asumimos una distribución sobre los datos y, a partir de ellos, estimamos el valor de los parámetros que definen esa distribución. Desde la estadística frecuentista, nos aproximamos a este concepto entendiendo que la distribución nos proporciona una idea sobre la frecuencia de aparición de cada dato. Sin embargo, desde la estadística bayesiana se interpreta la distribución como el grado de credibilidad de cada observación. Mientras que en el grado se desarrolla la interpretación frecuentista, en este capítulo comenzaremos recordándola para introducirnos en la interpretación bayesiana. Finalmente, a partir de esta, podremos introducir la inferencia variacional y sus principales propiedades.

De ahora en adelante y salvo que se especifique, trataremos con muestras aleatorias simples $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$ tomadas del espacio muestral $\mathcal{X}^N$. Llamaremos Θ a la familia de parámetros y $\hat{\boldsymbol{\theta}}$ al parámetro estimado, de manera que si tratamos con varios parámetros notaremos en general $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)$. y tomaremos $\Theta = \Theta_1 \times \cdots \times \Theta_n$, con $\Theta_i \subset \mathbb{R}, \forall i \in \{1, \dots, n\}$.

También notaremos, en general, para cada variable aleatoria X su función de masa de probabilidad o de densidad (según corresponda) como $p(x)$, y en caso de que haya duda entre distintas variables aleatorias notaremos $p_X(x)$.

1.1. Máxima verosimilitud

En estadística frecuentista, un método ampliamente extendido es el estimador máximo verosímil. Este trata de encontrar los parámetros que hacen más probables las observaciones analizadas.

Definición 1.1. Dada una observación $\mathbf{X} \in \mathcal{X}^N$, y su función masa de probabilidad (o densidad de probabilidad, si es continua) p, se define la **función de verosimilitud** asociada:

$$\begin{aligned} L_{\mathbf{X}} : \Theta &\longrightarrow \mathbb{R}_+^0 \\ \boldsymbol{\theta} &\longmapsto p(\mathbf{X}|\boldsymbol{\theta}). \end{aligned} \tag{1.1}$$

Definición 1.2. Llamamos **estimador máximo verosímil** o ***EMV*** y lo notamos $\hat{\boldsymbol{\theta}}^{\text{EMV}}$ o simplemente $\hat{\boldsymbol{\theta}}$ al que verifica:

$$L_{\mathbf{X}}(\hat{\boldsymbol{\theta}}) = \max_{\Theta}(L_{\mathbf{X}}(\boldsymbol{\theta})), \quad \forall \mathbf{X} \in \mathcal{X}^N \tag{1.2}$$

Podemos recordar de lo desarrollado durante el grado algunas propiedades interesantes de los estimadores de máxima verosimilitud:

- De haber un estadístico suficiente, $\hat{\theta}$ es función de él.

- De haber un estimador eficiente, sería $\hat{\theta}$ y en dicho caso sería el único estimador máximo verosímil.
- *(Teorema de invarianza de Zehna)* Si g es una función medible sobre Θ y $\hat{\theta}$ es el estimador máximo verosímil de θ, $g(\hat{\theta})$ es el estimador máximo verosímil de $g(\theta)$.

1.1.1. Ejemplo: Distribución categórica

Supongamos el siguiente experimento: los datos se clasifican en n categorías, $c_1, \ldots, c_n$, con probabilidades relativas $\pi_1, \ldots, \pi_n$ de manera exhaustiva, esto es, $\sum_{i=1}^{n} \pi_i = 1$. Más detalles sobre esta distribución, y todas las que encontramos a lo largo de este trabajo, pueden encontrarse en el Apéndice A.

Entonces, extraída una muestra aleatoria simple de tamaño N en la que se ha observado N_i veces la categoría c_i, el estimador de máxima verosimilitud de π_i es:

$$\hat{\pi}_i = \frac{N_i}{\sum_{j=1}^{n} N_j}.$$

Demostración. Sea $\Theta = [0,1]^n$, de manera que $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n) \in \theta$. Se tiene que dada una muestra aleatoria simple $\mathbf{X}$, la distribución $\mathbf{X}|\pi = \boldsymbol{\pi} \rightsquigarrow \text{Cat}(\boldsymbol{\pi})$. Así, calculamos el estimador máximo verosímil como sigue:

$$\begin{aligned}\hat{\pi}_i &= \max_{\Theta} \left(\prod_{j=1}^{N} p(\mathbf{x}|\boldsymbol{\pi}) \right) \\ &= \max_{\Theta} \left(\prod_{j=1}^{n} \pi_j^{N_j} \right).\end{aligned}$$

Por ser el logaritmo una función estrictamente creciente, el máximo de una función se alcanza en el mismo punto que el máximo de su logaritmo:

$$\log \left(\prod_{j=1}^{n} \pi_j^{N_j} \right) = \sum_{j=1}^{n} N_j \log(\pi_j).$$

Como además el logaritmo es derivable en $(0, +\infty)$, maximizamos por multiplicadores de Lagrange, restringiendo la condición $\sum_{j=1}^{n} \pi_j = 1$:

$$L = \sum_{j=1}^{n} N_j \log(\pi_j) + \lambda \left(\sum_{j=1}^{n} \pi_j - 1 \right).$$

Obtenemos:

$$\frac{\partial L}{\partial \pi_i} = \frac{N_i}{\pi_i} + \lambda = 0, \tag{1.3}$$

$$\frac{\partial L}{\partial \lambda} = \sum_{j=1}^{n} \pi_j - 1 = 0. \tag{1.4}$$

De la Ecuación 1.4 obtenemos la condición impuesta. Sin embargo, de la Ecuación 1.3,

sumando, obtenemos el valor de λ :

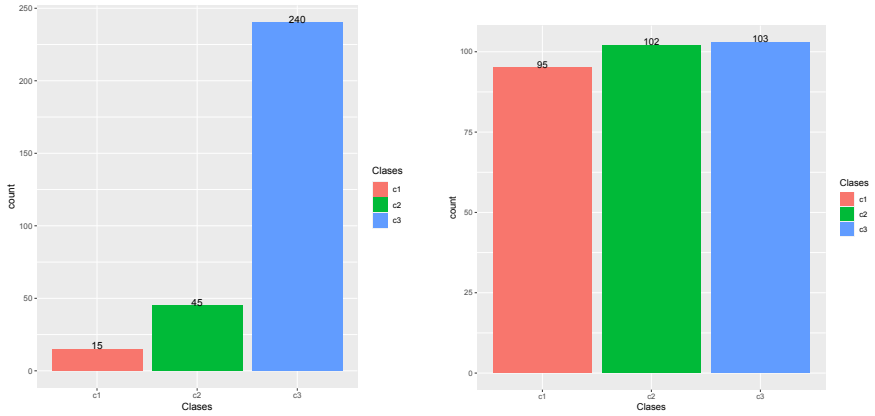
$$\sum_{j=1}^{n} N_j = \sum_{j=1}^{n} (\pi_j)\lambda, \tag{1.5}$$

por tanto $\lambda = \sum_{j=1}^{n} N_j$ y el punto crítico se alcanza cuando

$$\hat{\pi}_i^{\text{EMV}} = \frac{N_i}{\lambda} = \frac{N_i}{\sum_{j=1}^{n} N_j}. \tag{1.6}$$

Finalmente, como la matriz hessiana es diagonal con valores propios todos negativos, es definida negativa: lo que hemos obtenido es un máximo. □

Vemos ahora un ejemplo: consideremos dos variables categóricas con tres clases. En la Figura 1.1 se representa la distribución según cada clase. A partir de estas muestras, estimamos los valores de π_1, π_2 y π_3 en cada caso:



(a) $\pi_1 = 0.05, \pi_2 = 0.15, \pi_3 = 0.8$ (b) $\pi_1 = \pi_2 = \pi_3 = \frac{1}{3}$

Figura 1.1.: Diagrama de barras de una muestra n=300.

- En el panel (a) de la Figura 1.1, se describe una muestra de tamaño 300 de una distribución $\text{Cat}(0.05, 0.15, 0.8)$. Los estimadores máximo verosímiles son:
  - $\hat{\pi_1}^{\text{EMV}} = \frac{12}{300} = 0.04$
  - $\hat{\pi_2}^{\text{EMV}} = \frac{47}{300} \approx 0.157$
  - $\hat{\pi_3}^{\text{EMV}} = \frac{241}{300} \approx 0.803$
- En el panel (b) de la Figura 1.1, se describe una muestra de tamaño 300 de una distribución $\text{Cat}(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. Los estimadores máximo verosímiles son:
  - $\hat{\pi_1}^{\text{EMV}} = \frac{95}{300} \approx 0.317$
  - $\hat{\pi_2}^{\text{EMV}} = \frac{102}{300} = 0.34$
  - $\hat{\pi_3}^{\text{EMV}} = \frac{103}{300} \approx 0.343$

Dado que hemos tomado una muestra suficientemente rica, las estimaciones obtenidas son una buena aproximación de los parámetros de la distribución teórica.

1.2. Inferencia bayesiana

Como se mencionaba al comienzo del capítulo, en inferencia clásica o frecuentista se interpreta la probabilidad como una frecuencia en un período de tiempo que tiende a infinito. Así, los procedimientos que se desarrollan tienen que estar en concordancia con esta idea. Sin embargo, se puede interpretar la probabilidad como el grado de credibilidad de una observación. Esto permite añadir información subjetiva conocida u obtenida a priori para obtener un mayor grado de credibilidad. Es este nuevo enfoque el que se desarrolla mediante la inferencia bayesiana.

Definición 1.3. Dada una familia de distribuciones con un parámetro desconocido $\boldsymbol{\theta}$, llamamos **distribución a priori** de $\boldsymbol{\theta}$ a la distribución de probabilidad que asigna a cada $\boldsymbol{\theta}$ su grado de credibilidad $p(\boldsymbol{\theta})$.

Definición 1.4. Dada una muestra aleatoria simple de una distribución $\mathbf{X}$, llamamos **evidencia** al grado de credibilidad de una observación, $p(\mathbf{x})$.

Definición 1.5. Dada una familia de distribuciones con un parámetro desconocido $\boldsymbol{\theta}$ y una muestra aleatoria simple de dicha distribución $\mathbf{X}$, llamamos **distribución a posteriori** de $\boldsymbol{\theta}$ a la distribución de probabilidad $p(\boldsymbol{\theta}|\mathbf{x})$ que asigna a cada $\boldsymbol{\theta}$ su grado de credibilidad dada una observación $\mathbf{x}$.

Proposición 1.1. *La distribución a posteriori es proporcional al producto de la verosimilitud y la distribución a priori.*

Demostración. Basta con aplicar el teorema de Bayes:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}, \tag{1.7}$$

dado que $\mathbf{x}$ es observado, y por tanto, constante. □

Corolario 1.1. *Dada una observación $\mathbf{x} \in \mathcal{X}^N$, podemos calcular la evidencia $p(\mathbf{x})$ mediante:*

$$p(\mathbf{x}) = \int_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$$

Demostración. Integrando respecto de $\boldsymbol{\theta}$ en Ecuación 1.7:

$$\int_{\Theta} p(\boldsymbol{\theta}|\mathbf{x})\mathrm{d}\boldsymbol{\theta} = \int_{\Theta} \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}\mathrm{d}\boldsymbol{\theta}.$$

Luego:

$$1 = \int_{\Theta} \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}\mathrm{d}\boldsymbol{\theta}.$$

Como $p(\mathbf{x})$ no depende de $\boldsymbol{\theta}$, concluimos:

$$p(\mathbf{x}) = \int_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}.$$

□

Observación 1.1. El estimador puntual que se utiliza para inferencia bayesiana puede ser la media o la moda de la distribución a posteriori.

A continuación, vamos a calcular algunas distribuciones a posteriori básicas.

1.2.1. Ejemplo: Distribución binomial

Sea $X \rightsquigarrow \text{Bern}(\pi)$, y $x_1, \ldots, x_N$ observaciones de una muestra aleatoria simple de una distribución Bernoulli de parámetro π.

Consideremos una distribución uniforme a priori $\pi \rightsquigarrow \text{U}[0,1]$, esto es, el caso concreto $\beta(1,1)$:

$$p(\pi) = \begin{cases} 1 & \text{si } \pi \in [0,1], \\ 0 & \text{en caso contrario.} \end{cases}$$

Calculemos la distribución a posteriori, sea $S_N = \sum_{i=1}^N X_i$:

$$p(\pi)p(X_1, \ldots, X_N|\pi) = \pi^{S_N}(1-\pi)^{N-S_N} = \pi^{S_N+1-1}(1-\pi)^{N-S_N+1-1}.$$

Calculamos la evidencia:

$$\begin{aligned} p(X_1, \ldots, X_N) &= \int_0^1 p(\pi)p(X_1, \ldots, X_N|\pi)d\pi \\ &= \int_0^1 \pi^{S_N+1-1}(1-\pi)^{N-S_N+1-1}d\pi \\ &= \frac{\Gamma(S_N+1)\Gamma(N-S_N+1)}{\Gamma(N+2)}, \end{aligned}$$

donde hemos utilizado que tiene la forma de una distribución

$$\beta(\sum_{i=1}^N X_i + 1, N - \sum_{i=1}^N X_i + 1)$$

y por tanto, la integral vale 1.

Luego vemos que la distribución a posteriori es de la forma:

$$p(\pi|X_1, \ldots, X_N) = \frac{\Gamma(N+2)}{\Gamma(\sum_{i=1}^N X_i + 1)\Gamma(N - \sum_{i=1}^N X_i + 1)} \pi^{\sum_{i=1}^N X_i+1-1}(1-\pi)^{N-\sum_{i=1}^N X_i+1-1},$$

esto es, sigue una

$$\beta(\sum_{i=1}^N X_i + 1, N - \sum_{i=1}^N X_i + 1).$$

A partir de aquí, es sencillo repetir los cálculos para concluir que para una distribución a priori $\pi \rightsquigarrow \beta(u,v)$ se obtiene:

$$\pi|X \rightsquigarrow \beta(u + \sum_{i=1}^N X_i, v + N - \sum_{i=1}^N X_i).$$

Así, como se mencionaba, la uniforme no es más que el caso concreto $\beta(1,1)$.

Observación 1.2. Vemos que la distribución a priori y a posteriori son del mismo tipo: en este caso, las dos siguen una distribución beta. Cuando esto ocurre, decimos que la distribución a priori es **conjugada**.

1.3. Máximo a posteriori

Como estimación puntual, la más extendida es la moda de la distribución a posteriori. A este tipo de inferencia se le denomina máximo a posteriori. Veamos que es una generalización de la inferencia máximo verosímil:

Definición 1.6. Llamamos **estimación del máximo a posteriori** o ***MAP***, y lo notamos $\hat{\boldsymbol{\theta}}^{\text{MAP}}$ o simplemente $\hat{\boldsymbol{\theta}}$ al que verifica:

$$p(\hat{\boldsymbol{\theta}}|\mathbf{x}) = \max_{\Theta} p(\boldsymbol{\theta}|\mathbf{x}) = \max_{\Theta} \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}. \tag{1.8}$$

Por tanto, como la evidencia no depende del parámetro, se tiene:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

Proposición 1.2. *Si la distribución a priori es uniforme, la estimación del máximo a posteriori coincide con el estimador de máxima verosimilitud.*

Demostración. Si $p(\boldsymbol{\theta})$ es constante, entonces:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}) = \arg\max_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})$$

que es el estimador máximo verosímil. □

1.3.1. Ejemplo: Distribución binomial

Como veíamos en la Subsección 1.2.1, dada una distribución a priori $\pi \rightsquigarrow \beta(u, v)$, la distribución a posteriori es calculable. En concreto, para una muestra de tamaño N, la distribución a posteriori es

$$\beta(u + \sum_{i=1}^{N} X_i, v + N - \sum_{i=1}^{N} X_i).$$

El cálculo de la estimación máxima a posteriori no es más que el cálculo de la moda de la distribución a posteriori, que en nuestro caso es conocida, de manera que obtenemos:

$$\frac{\sum_{i=1}^{N} X_i + u - 1}{u + v + N - 2}.$$

Vemos que efectivamente, generaliza el caso de la distribución uniforme (donde $u = 1, v = 1$).

Tomemos una muestra y veamos qué información aportan las distintas a priori en cuanto a estimar mediante máxima verosimilitud o máximo a posteriori. Utilizaremos el experimento de lanzar una moneda, y una muestra donde hemos realizado 5 lanzamientos y los 5 han salido cara (éxito).

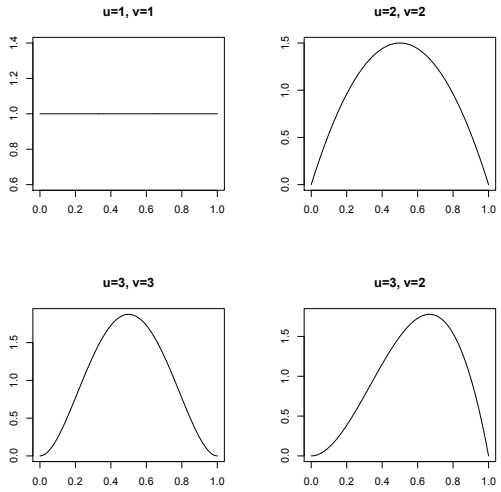


Figura 1.2.: Distintas distribuciones a priori utilizadas

- Cuando $u = 1, v = 1$ vemos que tenemos una distribución uniforme, esto es, obtendremos el estimador máximo verosímil: $\hat{\pi}^{\text{EMV}} = \frac{5}{5} = 1$. Sin embargo, parece poco realista que una moneda esté cargada de manera que siempre salga cara.
- Utilizamos la siguiente a priori, $u = 2, v = 2$. Entonces obtenemos:

$$\hat{\pi}^{\text{MAP}_2} = \frac{6}{7} \approx 0.8571$$

- Utilizamos la siguiente a priori, $u = 3, v = 3$. Entonces obtenemos:

$$\hat{\pi}^{\text{MAP}_3} = \frac{7}{9} \approx 0.7778$$

- Utilizamos la siguiente a priori, $u = 3, v = 2$. Entonces obtenemos:

$$\hat{\pi}^{\text{MAP}_4} = \frac{7}{8} \approx 0.875$$

Podemos ver que mediante el uso de una distribución a priori, se corrige la falta de credibilidad del estimador máximo verosímil. Además, como vemos en la Figura 1.2, los parámetros u y v son una medida de la concentración de los datos. Cuando son simétricos, se concentran entorno al valor 0.5. Esto es esperable del experimento aleatorio que hemos estudiado. Tomar un valor mucho mayor a 1 de u y v en la distribución a priori hace que las estimaciones MAP se concentren más entorno a un valor.

Esto es debido a su menor varianza. Al tomar valores menores a 1, la varianza disminuye cuanto menores sean los valores de u y v. Con esto, los datos se concentran entorno a 0 y 1. Podemos ver la abundancia de información que aporta la distribución a priori, sin perder la información que ya se tenía a partir de la inferencia clásica.

1.3.2. Aproximación bayesiana

Basándonos en la filosofía de la inferencia bayesiana, podemos hacernos la siguiente pregunta: ¿es posible obtener una predicción sin pasar por el cálculo de una estimación puntual? Es decir, ¿hay alguna forma de aprovechar toda la información de que se dispone para llegar a una predicción que la utilice? Hasta ahora, utilizábamos la estimación del parámetro $\boldsymbol{\theta}$ para obtenerla, que no era más que un resumen de la distribución a posteriori. Sin embargo, podemos trabajar con toda la información mediante la esperanza de las predicciones que obtendríamos, ponderada por el grado de credibilidad de cada parámetro. Así, obtenemos para cada muestra aleatoria simple $\mathbf{X}$:

$$p(\mathbf{x}^*|\mathbf{X}) = \int_\Theta p(\mathbf{x}^*|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{X})\mathrm{d}\boldsymbol{\theta} \tag{1.9}$$

Trabajamos así con una mayor abundancia de información.

Definición 1.7. Llamamos **densidad predictiva** a la función:

$$\begin{aligned} p : \mathcal{X} &\longrightarrow \mathbb{R}_0^+ \\ \mathbf{x}^* &\longmapsto p(\mathbf{x}^*|\mathbf{X}) \end{aligned} \tag{1.10}$$

Observación 1.3. Esto que parece una forma de proceder nueva, no se aleja demasiado de las expuestas con anterioridad, puesto que podemos expresar las predicciones que llevábamos a cabo como sigue:

$$\begin{aligned} p(\mathbf{x}^*|\hat{\theta}) &= \int_\Theta p(\mathbf{x}^*|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{X})\delta(\boldsymbol{\theta},\hat{\theta})\mathrm{d}\boldsymbol{\theta} \\ &= p(\mathbf{x}^*|\mathbf{X}) \end{aligned}$$

donde $\delta(x,y)$ es la delta de Kronecker y hemos utilizado la Ecuación 1.9.

1.4. Inferencia variacional

Hasta ahora, se ha hecho un planteamiento teórico del problema de inferencia. Lo que ocurre en la práctica es que la distribución a posteriori o la evidencia no suelen ser tratables. Por eso, se hace un planteamiento variacional del problema, buscando una aproximación de la distribución $p(\boldsymbol{\theta}|\mathbf{X})$. Para ello, se plantea un problema de optimización: se define un funcional, la divergencia Kullback-Leibler o KL-divergencia, que trataremos de minimizar obteniendo así una buena aproximación que se ajuste tanto a la distribución a priori como a las observaciones. Además, plantearemos un problema análogo: el de maximizar la cota inferior de la evidencia o ELBO.

Mediante $p(\mathbf{x})$ podemos saber el grado de credibilidad de la observación $\mathbf{x}$, esto es, la cantidad de información necesaria para aprender el valor $\mathbf{x}$. Para pasar esta información a bits, consideramos $-log_2(p(\mathbf{x}))$, donde se toma el signo opuesto para obtener un número de bits positivo o cero. Así, dado $p(\mathbf{x})$ podemos determinar cual es el número medio de bits necesarios para trabajar con toda la información que reside en la distribución de probabilidad.

Definición 1.8. Llamamos **entropía** de una distribución a la función:

$$H[\mathbf{x}] = -\int p(\mathbf{x})\ln(p(\mathbf{x}))\mathrm{d}\mathbf{x}.$$

En esta definición, por comodidad en el cálculo, se utiliza el logaritmo natural en lugar del logaritmo binario.

1.4.1. Divergencia Kullback-Leibler

Volviendo al problema original, disponemos de una distribución $p(\mathbf{x})$ que es desconocida y se quiere aproximar mediante $q(\mathbf{x})$. En este caso, podríamos medir la información media adicional necesaria para explicar $p(\mathbf{x})$ a partir de $q(\mathbf{x})$. Para ello, basta con restarle a la información que explica $q(\mathbf{x})$ sobre $p(\mathbf{x})$ la propiamente necesaria para explicar $p(\mathbf{x})$, esto es:

$$\begin{aligned}\mathrm{KL}(p||q) &= -\int_{\mathcal{X}} p(\mathbf{x}) \ln(q(\mathbf{x}))\mathrm{d}\mathbf{x} - \left(-\int_{\mathcal{X}} p(\mathbf{x}) \ln(p(\mathbf{x})\mathrm{d}\mathbf{x}\right) \\ &= -\int_{\mathcal{X}} p(\mathbf{x}) \ln\left(\frac{q(\mathbf{x})}{p(\mathbf{x})}\right)\mathrm{d}\mathbf{x}\end{aligned}$$

Definición 1.9. Llamamos entropía relativa de p respecto a q, **divergencia Kullback-Leibler** o abreviadamente **KL-divergencia** a:

$$\mathrm{KL}(p||q) = -\int_{\mathcal{X}} p(\mathbf{x}) \ln\left(\frac{q(\mathbf{x})}{p(\mathbf{x})}\right)\mathrm{d}\mathbf{x}$$

Utilizaremos una desigualdad que podemos ver demostrada en [CL18] para establecer las propiedades de la KL divergencia:

Proposición 1.3. *(Desigualdad de Jensen) Si f es una función convexa y p es una función de densidad o de masa de probabilidad, entonces:*

$$\mathbb{E}[f(p(\mathbf{x}))] \geq f(\mathbb{E}[p(\mathbf{x})])$$

y se da la igualdad si y solo si f es estrictamente convexa.

Proposición 1.4. *Dadas dos distribuciones p y q, entonces $KL(p||q) \geq 0$ y*

$$KL(p||q) = 0 \iff p = q$$

Demostración. Sea $\mathcal{X}$ el soporte de p, $\hat{\mathcal{X}}$ el soporte de q :

$$\begin{aligned}-\mathrm{KL}(p||q) &= \int_{\mathcal{X}} p(\mathbf{x}) \ln\left(\frac{q(\mathbf{x})}{p(\mathbf{x})}\right)\mathrm{d}\mathbf{x} \\ &\leq \ln\left(\int_{\mathcal{X}} p(\mathbf{x})\frac{q(\mathbf{x})}{p(\mathbf{x})}\mathrm{d}\mathbf{x}\right) \qquad (1.11) \\ &= \ln\left(\int_{\mathcal{X}} q(\mathbf{x})\mathrm{d}\mathbf{x}\right) \\ &\leq \left(\int_{\hat{\mathcal{X}}} q(\mathbf{x})\mathrm{d}\mathbf{x}\right) \\ &= \ln 1 \\ &= 0\end{aligned}$$

donde hemos utilizado en 1.11 que la función logaritmo es estrictamente cóncava, y por tanto verifica la desigualdad de Jensen. Queda probado que la KL divergencia es siempre no

negativa.

Estudiemos ahora cuándo es nula. En primer lugar, la primera desigualdad se obtiene si $\frac{p(\mathbf{x})}{q(\mathbf{x})}$ es constante salvo un conjunto de medida nula y por tanto:

$$1 = \int_{\mathcal{X}} p(\mathbf{x}) = K \int_{\mathcal{X}} q(\mathbf{x}).$$

Por otro lado, la segunda igualdad se da si:

$$\int_{\mathcal{X}} q(\mathbf{x}) = \int_{\hat{\mathcal{X}}} q(\mathbf{x}) = 1,$$

y por tanto $K = 1$, esto es, $q = p$ salvo un conjunto de medida nula. □

Vemos que la divergencia Kullback-Leibler no funciona de manera simétrica, esto es, para dos distribuciones distintas:

$$\mathrm{KL}(q||p) \neq \mathrm{KL}(p||q).$$

Así, la pregunta a plantear surge de manera natural: ¿qué debemos minimizar, $\mathrm{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X}))$ o $\mathrm{KL}(p(\boldsymbol{\theta}|\mathbf{X})||q(\boldsymbol{\theta}))$?

Definición 1.10. Llamamos proyección de información, **I-proyección** o **KL-inversa** a

$$\mathrm{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X})) = \int_{\Theta} q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathbf{X})}) \mathrm{d}\boldsymbol{\theta}. \tag{1.12}$$

Observación 1.4. Cuando $p(\boldsymbol{\theta}|\mathbf{X})$ es pequeña, eso fuerza a que una buena aproximación q tenga que ser también pequeña.

Definición 1.11. Llamamos proyección de momentos, **M-proyección** o **KL-directa** a

$$\mathrm{KL}(p(\boldsymbol{\theta}|\mathbf{X})||q(\boldsymbol{\theta})) = \int_{\Theta} p(\boldsymbol{\theta}|\mathbf{X}) \ln(\frac{p(\boldsymbol{\theta}|\mathbf{X}))}{q(\boldsymbol{\theta})}) \mathrm{d}\boldsymbol{\theta}. \tag{1.13}$$

Observación 1.5. Cuando $p(\boldsymbol{\theta}|\mathbf{X}) > 0$ si $q(\boldsymbol{\theta}) = 0$ entonces la M-proyección diverge a infinito.

Así, podemos entender la I-proyección como una medida de la información que se pierde al explicar $p(\boldsymbol{\theta}|\mathbf{x})$ mediante una distribución $q(\boldsymbol{\theta})$, mientras que podemos interpretar la M-proyección como la cantidad de información que alberga $q(\boldsymbol{\theta})$ sobre $p(\boldsymbol{\theta}|\mathbf{x})$. Por tanto, parece razonable la decisión de intentar minimizar la I-proyección para buscar una buena aproximación. Sin embargo, este resultado no es útil a efectos prácticos puesto que depende de la distribución $p(\boldsymbol{\theta}|\mathbf{x})$, que era intratable. Intentemos encontrar un funcional equivalente que minimizar de cuya información dispongamos.

Proposición 1.5. *En las condiciones planteadas, se verifica:*

$$\arg\min_{q \in \mathcal{Q}} KL(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X})) = \arg\min_{q \in \mathcal{Q}} \int_{\Theta} q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})}) d\boldsymbol{\theta}$$

Demostración. Ahora, utilicemos $p(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\mathbf{X},\boldsymbol{\theta})}{p(\mathbf{X})}$, de manera que obtenemos:

$$\begin{aligned}
\mathrm{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X})) &= \int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathbf{X})}) d\boldsymbol{\theta} \\
&= \int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})p(\mathbf{X})}) d\boldsymbol{\theta} \\
&= \int_\Theta q(\boldsymbol{\theta}) \left(\ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}) - \ln(p(\mathbf{X}) \right) d\boldsymbol{\theta} \\
&= \int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}) d\boldsymbol{\theta} - \int_\Theta q(\boldsymbol{\theta}) \ln(p(\mathbf{X})) d\boldsymbol{\theta} \\
&= \int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}) d\boldsymbol{\theta} - \ln(p(\mathbf{X}))
\end{aligned}$$

Como $p(\mathbf{X})$ no depende de $\boldsymbol{\theta}$ los dos funcionales alcanzan su mínimo a la vez. □

Por tanto, hemos encontrado un funcional equivalente que minimizar:

$$\int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}) d\boldsymbol{\theta}.$$

Este funcional depende exclusivamente de información conocida: la distribución conjunta. Así, hemos conseguido encontrar una solución práctica para el problema de aproximación de la distribución a posteriori.

1.4.1.1. Divergencia Kullback-Leibler entre dos normales univariantes

Veamos ahora un ejemplo. Tomemos dos distribuciones normales univariantes: $X \rightsquigarrow \mathcal{N}(\mu_1, \sigma_1)$ y $\tilde{X} \rightsquigarrow \mathcal{N}(\mu_2, \sigma_2)$. Sea $p(x)$ la función de densidad de probabilidad de X y $q(x)$ la de $\tilde{X}$:

$$\begin{aligned}
\mathrm{KL}(p||q) &= -\int_{\mathbb{R}} p(x) \ln(q(x)) \mathrm{d}x - \left(-\int_{\mathbb{R}} p(x) \ln p(x) \mathrm{d}x \right) \\
&= \int_{\mathbb{R}} p(x) (\ln(p(x)) - \ln q(x)) \mathrm{d}x.
\end{aligned}$$

En primer lugar,

$$\ln p(x) = -\frac{1}{2}\log(2\pi) - \log(\sigma_1) - \frac{1}{2}\left(\frac{x-\mu_1}{\sigma_1}\right)^2$$

$$\ln q(x) = -\frac{1}{2}\log(2\pi) - \log(\sigma_2) - \frac{1}{2}\left(\frac{x-\mu_2}{\sigma_2}\right)^2.$$

Luego:

$$\ln p(x) - \ln q(x) = \ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2}\left[\left(\frac{x-\mu_2}{\sigma_2}\right)^2 - \left(\frac{x-\mu_1}{\sigma_1}\right)^2\right].$$

Por lo tanto, tenemos que calcular:

$$\mathbb{E}_p\left[\ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2}\left\{\left(\frac{X-\mu_2}{\sigma_2}\right)^2 - \left(\frac{X-\mu_1}{\sigma_1}\right)^2\right\}\right].$$

$$\begin{aligned}
\mathrm{KL}(p||q) &= \mathbb{E}_p\left[\ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2}\left\{\left(\frac{X-\mu_2}{\sigma_2}\right)^2 - \left(\frac{X-\mu_1}{\sigma_1}\right)^2\right\}\right] \\
&= \mathbb{E}_p\left[\ln\left(\frac{\sigma_2}{\sigma_1}\right)\right] + \mathbb{E}_p\left[\frac{1}{2}\left\{\left(\frac{X-\mu_2}{\sigma_2}\right)^2 - \left(\frac{X-\mu_1}{\sigma_1}\right)^2\right\}\right] \\
&= \ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2\sigma_2^2}\mathbb{E}_p\left[(X-\mu_2)^2\right] - \frac{1}{2\sigma_1^2}\mathbb{E}_p\left[(X-\mu_1)^2\right] \\
&= \ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2\sigma_2^2}\mathbb{E}_p\left[(X-\mu_2)^2\right] - \frac{1}{2}.
\end{aligned}$$

Ahora, utilizamos lo siguiente:

$$(X-\mu_2)^2 = (X-\mu_1+\mu_1-\mu_2)^2 = (X-\mu_1)^2 + 2(X-\mu_1)(\mu_1-\mu_2) + (\mu_1-\mu_2)^2.$$

Luego:

$$\begin{aligned}
\mathbb{E}_p\left[(X-\mu_2)^2\right] &= \mathbb{E}_p\left[(X-\mu_1)^2\right] + \mathbb{E}_p\left[2(X-\mu_1)(\mu_1-\mu_2)\right] + \mathbb{E}_p\left[(\mu_1-\mu_2)^2\right] \\
&= \sigma_1^2 + 2(\mu_1-\mu_2)\mathbb{E}_p\left[(X-\mu_1)\right] + (\mu_1-\mu_2)^2 \\
&= \sigma_1^2 + (\mu_1-\mu_2)^2.
\end{aligned}$$

Finalmente, sustituimos:

$$\mathrm{KL}(p||q) = \ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2}.$$

Fijemos en $\tilde{X} \rightsquigarrow \mathcal{N}(0,1)$. Veamos las distintas distancias a otras normales $X \rightsquigarrow \mathcal{N}(\mu_1, \sigma_1)$:

$$\mathrm{KL}(p||q) = \ln\left(\frac{1}{\sigma_1}\right) + \frac{\sigma_1^2 + \mu_1^2}{2} - \frac{1}{2}. \tag{1.14}$$

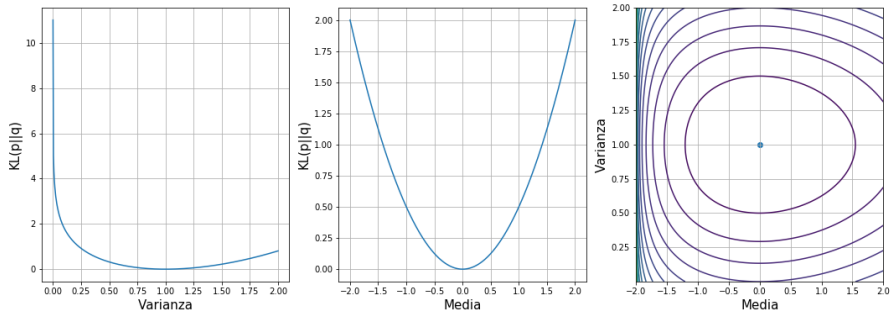


Figura 1.3.: Representación de la KL divergencia de las distribuciones $\mathcal{N}(\mu_1, \sigma_1)$ a una $\mathcal{N}(0,1)$. En primer lugar, $\mu_1 = 0$ y σ_1 es desconocido. En segundo lugar μ_1 es desconocido y $\sigma_1 = 1$. En tercer lugar, se representan las líneas de contorno cuando μ_1 y σ_1 son desconocidos.

- En primer lugar, estudiamos el caso en que la media es conocida y coincide con la distribución fijada en Ecuación 1.14. Como vemos en la Figura 1.3 la divergencia no es simétrica respecto de la varianza de la distribución fija, $\sigma = 1$. Sin embargo, esto era esperable.

  Al aproximar una varianza $\sigma = 1$ por $\sigma_1 = 0.01$, se escoge una varianza que es 100 veces menor. Por otro lado, al aproximarla por $\sigma = 1.99$, la varianza se aproxima por una que es aproximadamente 2 veces mayor. Así, mientras que la distancia usual hubiese determinado un error simétrico en ambas aproximaciones, la KL divergencia lo corrige.

- En segundo lugar, estudiamos el caso en que la varianza es conocida y coincide con la de la distribución fijada en Ecuación 1.14. Como vemos en la Figura 1.3 la divergencia es simétrica respecto de la media de la distribución fija, $\mu = 0$.

  De hecho, coincide con la distancia medida respecto a la norma euclídea. En este caso es coherente. Una media $\mu = 0$ está igualmente aproximada por una $\mu_1 = 1$ que por una $\mu_1 = -1$. Por tanto, cabe esperar que sendas distribuciones sean aproximaciones igual de buenas.

- Finalmente, en las líneas de contorno de la Figura 1.3 vemos con claridad como estas dos apreciaciones se manifiestan de manera simúltanea: mientras que hay simetría vertical, provocada por las aproximaciones de la media, se aprecia una asimetría horizontal causada por las aproximaciones de la varianza.

Como conclusión, este ejemplo pone de manifiesto que la KL divergencia es una buena medida de cuán buena es la aproximación de una distribución de probabilidad por otra, recogiendo la información de cada parámetro de manera coherente.

1.4.2. Cota inferior de la evidencia

Desde la inferencia bayesiana, se aborda la estimación a partir del cálculo de la evidencia, como veíamos al probar la Proposición 1.1. Sin embargo, en la práctica, es muy común que la evidencia no sea tratable. Para abordar este problema, utilizaremos la KL divergencia para obtener una cota inferior:

$$0 \leq \mathrm{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X})) = \int_{\Theta} q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})}) d\boldsymbol{\theta} + \ln(p(\mathbf{X})).$$

Luego,

$$\ln(p(\mathbf{X})) \geq -\int_{\Theta} q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})}) d\boldsymbol{\theta}.$$

Así, hemos encontrado una cota inferior de la evidencia.

Definición 1.12. Llamamos **cota inferior de la evidencia** o ***ELBO*** al siguiente funcional:

$$\mathrm{ELBO}(q) = -\int_{\Theta} q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})}) d\boldsymbol{\theta}. \tag{1.15}$$

Proposición 1.6.

1. $\mathbb{E}_q[p(\mathbf{X}|\boldsymbol{\theta})] = ELBO(q) + KL(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}))$.

2. $\ln(p(\mathbf{X})) = ELBO(q) + KL(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X}))$.

Demostración. Repetimos lo anterior al contrario, de manera que obtenemos:

$$\begin{aligned}\text{ELBO}(q) &= -\int_\Theta q(\boldsymbol{\theta}) \ln\left(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta} \\ &= -\int_\Theta q(\boldsymbol{\theta}) \ln\left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})p(\mathbf{X}|\boldsymbol{\theta}}\right) \mathrm{d}\boldsymbol{\theta} \\ &= -\int_\Theta q(\boldsymbol{\theta}) \ln\left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta} + \int_\Theta q(\boldsymbol{\theta}) \ln(p(\mathbf{X}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \\ &= \text{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta})) + \mathbb{E}_q[p(\mathbf{X}|\boldsymbol{\theta})].\end{aligned} \tag{1.16}$$

Por otro lado:

$$\text{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X})) = \int_\Theta q(\boldsymbol{\theta}) \ln(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X},\boldsymbol{\theta})}) d\boldsymbol{\theta} + \ln(p(\mathbf{X})) = -\text{ELBO}(q) + \ln(p(\mathbf{X})).$$

□

Así, la Proposición 1.6 en el apartado 2 pone de manifiesto el comportamiento constante de la evidencia respecto de la KL divergencia y la cota inferior, ambas dependientes variacionalmente de la distribución aproximada q. Esto permite replantear el problema, puesto que la misma solución que minimiza la KL divergencia ha de maximizar la ELBO, y por tanto:

$$q(\boldsymbol{\theta}) = \arg\min_{q\in\mathcal{Q}}(\text{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X}))) = \arg\max_{q\in\mathcal{Q}}(\text{ELBO}(q(\boldsymbol{\theta})).$$

Para ello tomaremos una familia de funciones $\mathcal{Q}$ lo suficientemente amplia como para obtener una buena aproximación de la distribución $p(\boldsymbol{\theta}|\mathbf{X})$ pero a la vez lo suficientemente sencilla para que el problema sea tratable. Así queda solucionado el problema entre $\text{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\mathbf{X}))$ y la evidencia: basta con tratar en su lugar con la maximización de la cota inferior de la evidencia, cuya solución es equivalente y tratable.

Observación 1.6. Analicemos la relación $\text{ELBO}(q) = \mathbb{E}[\ln(p(\mathbf{X}|\boldsymbol{\theta})] - \text{KL}[q(\boldsymbol{\theta})||p(\boldsymbol{\theta})]$:

- El término $\mathbb{E}[\ln(p(\mathbf{X}|\boldsymbol{\theta})]$ no es más que el logaritmo de la verosimilitud. Esto hace que se seleccionen distribuciones $q(\boldsymbol{\theta})$ que expliquen bien los datos observados.
- El término $-\text{KL}[q(\boldsymbol{\theta})||p(\boldsymbol{\theta})]$ indica la KL-divergencia entre la distribución a priori de $\boldsymbol{\theta}$ y la distribución aproximada. Esto hace que la distribución seleccionada esté cerca de la distribución a priori.

Luego el problema mencionado pone de manifiesto el equilibrio entre ajustarse a las observaciones y a la distribución a priori, propia del enfoque bayesiano.

1.4.2.1. Cota inferior de la evidencia para la aproximación *mean field*

Finalmente, para tratar con un ejemplo de cálculo de la cota inferior de la evidencia, se desarrollarán las restricciones necesarias para obtener una aproximación de la distribución latente: la aproximación ***mean field***. Esta aproximación en el sentido de la KL-divergencia se desarrolla dentro de la familia de distribuciones con marginales independientes. Esto nos dará una distribución que es computacionalmente más tratable.

Así, buscamos una distribución:

$$q(\mathbf{X}) = \prod_{i=1}^{n} q(\mathbf{X}_i)$$

de manera que maximice ELBO(q). Por sí misma, la familia *mean field* no tiene ninguna conexión con los datos observados. Es el planteamiento del problema mediante el ELBO(q) lo que hará que seleccionemos la distribución que mejor se ajuste a la evidencia. Por otro lado, también cabe destacar que no se le impone ninguna distribución en concreto a cada marginal. Por lo tanto, recoge toda la información acerca de cada marginal, si bien no captura la correlación entre ellas. Estos son los principales puntos fuertes y débiles de este método.

Proposición 1.7. *La mejor aproximación mean field verifica:*

$$\ln(q_j^*(\theta_j)) = \ln(\mathbb{E}_{i\neq j}[p(\mathbf{X}, \boldsymbol{\theta})]) + K,$$

donde K es una constante y $\mathbb{E}_{i\neq j}[p(\mathbf{X}, \boldsymbol{\theta})]$ es la esperanza de la distribución conjunta respecto del resto de marginales.

Demostración. Expresemos la cota inferior de la evidencia para esta aproximación, sea $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)$:

$$\begin{aligned}
\text{ELBO}[q] &= -\int_\Theta q(\boldsymbol{\theta}) \ln\left(\frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta} \\
&= -\int_\Theta \prod_{i=1}^{m} q_i(\theta_i) \ln\left(\frac{\prod_{i=1}^{m} q_i(\theta_i)}{p(\mathbf{X}, \boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta} \\
&= -\int_\Theta \prod_{i=1}^{m} q_i(\theta_i) \sum_{i=1}^{m} \ln\left(\frac{q_i(\theta_i)}{p(\mathbf{X}, \boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta} \\
&= \int_\Theta \ln(p(\mathbf{X}, \boldsymbol{\theta})) \prod_{i=1}^{m} q_i(\theta_i)\mathrm{d}\theta_i - \sum_{j=1}^{m} \int_\Theta \ln(q_j(\theta_j)) \prod_{i=1}^{m} q_i(\theta_i)\mathrm{d}\theta_i \\
&= \int_{\Theta_j} q_j(\theta_j) \left(\int_{\Theta'} \ln(p(\mathbf{X}, \boldsymbol{\theta})) \prod_{i=1_{i\neq j}}^{m} q_i(\theta_i)\mathrm{d}\theta_i \right) - \sum_{j=1}^{m} \int_{\Theta'} \ln(q_j(\theta_j)) q_j(\theta_j)\mathrm{d}\theta_j + const. \\
&= \int_{\Theta_j} q_j(\theta_j) \ln(\tilde{p}(\mathbf{X}, \theta_j))\mathrm{d}\theta_j - \int_{\Theta_j} q_j(\theta_j) \ln q_j(\theta_j)\mathrm{d}\theta_j + const. \\
&= \int_{\Theta_j} q_j(\theta_j) \ln\left(\frac{\tilde{p}(\mathbf{X}, \theta_j)}{q_j(\theta_j)}\right) \mathrm{d}\theta_j + const. \\
&= -\mathrm{KL}(q_j(\theta_j) || \tilde{p}(\mathbf{X}, \theta_j)) + const.
\end{aligned}$$

Donde

$$ln(\tilde{p}(\mathbf{X}, \theta_j)) = \int_\Theta \ln(p(\mathbf{X}, \boldsymbol{\theta})) \prod_{i=1_{i\neq j}}^{m} q_i(\theta_i)\mathrm{d}\theta_i + const.,$$

y

$$\Theta' = \Theta_1 \times \cdots \times \hat{\Theta}_j \times \cdots \times \Theta_m.$$

Por las propiedades que veíamos de la KL-divergencia, si dejamos fijo el término q_j el mínimo de $\mathrm{KL}(q_j(\theta_j) || \tilde{p}(\mathbf{X}, \theta_j))$ se alcanza cuando $q_j = \tilde{p}(\mathbf{X}, \theta_j)$. Por tanto, de encontrar dicho máximo,

esto es, la aproximación que buscábamos, debería verificar $q^*(\boldsymbol{\theta}) = \prod_{j=1}^{m} q_j^*(\theta_j)$ con:

$$\ln(q_j^*(\theta_j)) = \ln\left(\int_{\Theta'} \ln(p(\mathbf{X}, \boldsymbol{\theta})) \prod_{i=1_{i\neq j}}^{m} q_i(\theta_i) \mathrm{d}\theta_i\right) + const.$$

Vale la pena fijarse en que:

$$\int_{\Theta'} \ln(p(\mathbf{X}, \boldsymbol{\theta})) \prod_{i=1_{i\neq j}}^{m} q_i(\theta_i) \mathrm{d}\theta_i = \mathbb{E}_{i\neq j}[p(\mathbf{X}, \boldsymbol{\theta})], \tag{1.17}$$

que no es más que el calculo de la esperanza de la distribución conjunta respecto del resto de las marginales, $q_i(\theta_i)$. □

La constante se determina mediante normalización y se suele calcular en la práctica, si se requiere, posteriormente.

Cada una de estas condiciones depende del resto de distribuciones, y por tanto no nos ofrecen un máximo, sino que nos proveen de m ecuaciones restrictivas en las que se enmarca la futura aproximación, llevada a cabo por métodos numéricos. Para ello se desarrolla el algoritmo *CAVI*, que de manera iterativa, fija una de las componentes para optimizar las demás. Un estudio más pormenorizado de este problema puede encontrarse en [MKB17].

2. Procesos gaussianos

Así como se afronta la inferencia desde la perspectiva clásica y la bayesiana, encontramos sendos planteamientos en la solución del modelo lineal general. Mientras que el punto de partida es el mismo en ambos problemas, surge un concepto que abre nuevas opciones desde el enfoque bayesiano: las funciones base. Mediante ellas, conseguiremos proyectar las observaciones a espacios de mayor dimensión para una modelización más compleja de los datos. Los procesos gaussianos generalizan la distribución normal o gaussiana mutivariante. Están completamente determinadas por una función media y una función núcleo o de covarianza. En la función de covarianza encontraremos una dependencia con las funciones base, ofreciéndose así la interpretación de los procesos gaussianos como distribuciones sobre funciones. A partir de este nuevo punto de vista, revisaremos el concepto de regresión lineal. A su conclusión, a partir del estudio de la regresión lineal se introducirá el problema de la clasificación mediante procesos gaussianos.
En este capítulo se ha utilizado especialmente el paquete *sklearn* del lenguaje de programación python para la elaboración de las figuras. Los códigos utilizados para la obtención de las figuras de este trabajo pueden encontrarse en el Apéndice B.

2.1. Modelo lineal general clásico

La teoría de modelos lineales sienta las bases para tratar con problemas en los que se estudia el comportamiento de una variable aleatoria a partir de otras. Así, se establecen unos parámetros sobre los que se hace inferencia para tratar de discernir dichas relaciones. Para ello, se establecen criterios adecuados para la resolución eficaz del problema: el Modelo Lineal General.

Definición 2.1. Un **modelo lineal** es una expresión de la forma $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, donde:

- $\mathbf{Y}$ es un vector observable de dimensión N.
- $\mathbf{X}$ es una matriz de dimensión $N \times k$ con $k < N$ llamada matriz de diseño, con datos observables.
- $\boldsymbol{\beta}$ es un vector de dimensión k. Cada componente β_j pondera la influencia de la columna j-ésima de $\mathbf{X}$ en $\mathbf{Y}$, por lo que recibe el nombre de vector de efectos. Es desconocido y no observable.
- $\boldsymbol{\varepsilon}$ es un vector de dimensión N desconocido y no observable denominado vector de errores o ruido.

$$\begin{pmatrix} Y_1 \\ \vdots \\ Y_N \end{pmatrix} = \begin{pmatrix} X_{11} & \cdots & X_{1k} \\ \vdots & \vdots & \vdots \\ X_{N1} & \cdots & X_{Nk} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{pmatrix}. \tag{2.1}$$

Los modelos se clasifican según si el vector de efectos es fijo o aleatorio y según si el rango de $\mathbf{X}$ es máximo o no. En el caso de efectos fijos y matriz de diseño observable, que es el tratado en el desarrollo de la asignatura Inferencia Estadística, los factores estocásticos que intervienen en el comportamiento de $\mathbf{Y}$ están descritos mediante el ruido $\boldsymbol{\varepsilon}$.

2.1.1. Regresión lineal simple

Desde el enfoque de la inferencia clásica, estudiamos durante el grado la regresión lineal simple, que repasaremos en esta sección.

Definición 2.2. Dado un modelo lineal general $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, decimos que es un **modelo de Gauss-Markov** si verifica:

$$\mathbb{E}[\boldsymbol{\varepsilon}] = 0, \qquad \mathbb{E}[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T] = \sigma^2 \mathbf{I}_N, \tag{2.2}$$

con $\sigma \in \mathbb{R}$.

En la estimación de este modelo, se trata de inferir el vector de efectos $\boldsymbol{\beta}$ y la varianza σ^2. Para ello, se parte de la hipótesis de normalidad: $\boldsymbol{\varepsilon} \rightsquigarrow \mathcal{N}(0, \sigma^2 \mathbf{I}_N)$. Esto no es más que decir que

$$\mathbf{Y} \rightsquigarrow \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_N). \tag{2.3}$$

Llegados a este punto, buscamos el estimador máximo verosímil. La función de verosimilitud, $\forall y \in \mathbb{R}^N$, es:

$$L_{\mathbf{y}}(\boldsymbol{\beta}, \sigma^2) = \frac{1}{(2\pi)^{n/2}\sigma^n} \exp\left(-\frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2}{2\sigma^2}\right). \tag{2.4}$$

Vemos que:

$$\arg\max_{\boldsymbol{\beta}} L_{\mathbf{y}}(\boldsymbol{\beta}, \sigma^2) = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2, \tag{2.5}$$

luego el estimador máximo verosímil de $\boldsymbol{\beta}$ resulta de hacer una estimación por mínimos cuadrados.

Proposición 2.1. *El estimador por mínimos cuadrados del vector de efectos $\boldsymbol{\beta}$ existe y es único en modelos de rango máximo, en cuyo caso es:*

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}.$$

Demostración. Tratamos de minimizar la función $\mathrm{SE}(\boldsymbol{\beta}) = \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2$. Para ello derivamos e igualamos a cero:

$$\frac{\partial \mathrm{SE}}{\partial \beta_h} = -2\sum_{i=1}^{N}\left(Y_i \sum_{j=1}^{k} x_{ij}\boldsymbol{\beta}_j\right)x_{ih} = 0 \tag{2.6}$$

Luego,

$$\sum_{i=1}^{N} Y_i x_{ih} = \sum_{i=1}^{N}\sum_{j=1}^{k} x_{ij}x_{ih}\beta_j, \tag{2.7}$$

que no es más que decir $\mathbf{X}^T\mathbf{Y} = \mathbf{X}^T\mathbf{X}\boldsymbol{\beta}$. Así, como $\mathbf{X}$ es de rango máximo:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{2.8}$$

□

Una vez estimado el vector de efectos, buscamos la estimación máximo verosímil de la varianza.

Proposición 2.2. *El estimador máximo verosímil de la varianza σ^2 es:*

$$\hat{\sigma}^2 = \frac{\left\|\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right\|^2}{N}.$$

Demostración. Basta con tomar el logaritmo de la verosimilitud:

$$\ln L_{\mathbf{y}}(\boldsymbol{\beta}, \sigma^2) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2}{2\sigma^2},$$

de manera que por ser el logaritmo una función estrictamente monótona, derivando e igualando a cero encontramos el estimador:

$$\begin{aligned}\frac{\partial \ln L_{\mathbf{y}}(\boldsymbol{\beta}, \sigma^2)}{\partial \sigma^2} &= -\frac{N}{2}\left(\frac{2\pi}{2\pi\sigma^2}\right) + \frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2}{2\sigma^4} \\ &= -\frac{N}{2\sigma^2} + \frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2}{2\sigma^4} \\ &= 0.\end{aligned}$$

Así,

$$\hat{\sigma}^2 = \frac{\left\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right\|^2}{N}.$$

□

Ejemplo 2.1. La base de datos iris es ampliamente conocida por su uso en el aprendizaje de estadística computacional y *machine learning*. Los datos responden a características de tres especies de flores distintas: setosa, virginica y versicolor. Estas características son la longitud y anchura del sépalo y el pétalo de cada flor, medida en centímetros. En la base de datos encontramos un total de 150 registros, repartidos de manera casi uniforme en las tres especies distintas. Llevaremos a cabo las estimaciones máximo verosímiles para la recta de regresión entre las variables anchura del sépalo y longitud del pétalo de dicha base de datos. Por tanto, contamos con una matriz de diseño de tamaño 150×2. Podemos ver el resultado en la Figura 2.1. Sombreada, encontramos la recta con una región que comprende dos veces la desviación típica del ruido. Vemos que, excepto unos pocos de datos, los datos caen dentro de esta región, por lo que la recta de regresión obtenida es capaz de predecir con una incertidumbre adecuada.

2.2. Modelo lineal general bayesiano

De manera análoga a la desarrollada en el Capítulo 1, exponemos el mismo problema desde el punto de vista de la inferencia bayesiana. Para ello, partiremos de una distribución a priori sobre el vector de efectos. Se tratará de determinar la función a posteriori de $\mathbf{Y}$, para conseguir así una estimación máximo a posteriori.

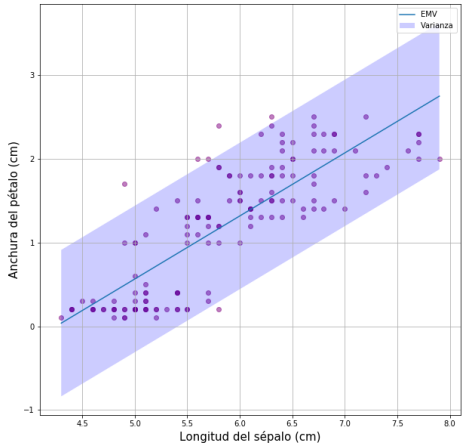
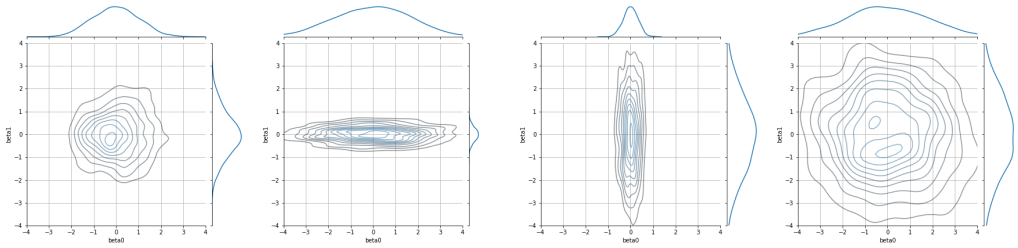


Figura 2.1.: Recta de regresión estimada por máxima verosimilitud de la anchura del pétalo respecto de la longitud del sépalo de la base de datos iris. Sombreado, los valores de la recta de regresión $\pm 2\sigma$.

En primer lugar, las condiciones de partida son las de la Ecuación 2.3. Además, tomaremos como distribución a priori $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0, \boldsymbol{\Sigma})$.

Ejemplo 2.2. Podemos ver en la Figura 2.2 algunos ejemplos de distribuciones a priori, y como influye su elección en la información que aportamos a la estimación.

Para una regresión lineal simple, al tomar una distribución como la representada en el panel (b), se aporta una certeza del 95 % a una pendiente que esté en el intervalo $[-3.4, 3.4]$ y una ordenada en el origen que esté en $[-0.6, 0.6]$.



(a) $\sigma_0^2 = 1, \sigma_1^2 = 1$ (b) $\sigma_0^2 = 3, \sigma_1^2 = 0.1$ (c) $\sigma_0^2 = 0.1, \sigma_1^2 = 3$ (d) $\sigma_0^2 = 3, \sigma_1^2 = 3$

Figura 2.2.: Distribuciones normales bivariantes representadas por los mapas de contorno de sus funciones de densidad, con función de densidad de las marginales al margen. Se ha tomado la media cero en los cuatro casos, y una matriz de covarianzas diagonal, cuyos valores se indican bajo cada uno de los paneles.

Ahora, podemos aplicar el teorema de Bayes para obtener la distribución a posteriori:

$$p(\boldsymbol{\beta}|\mathbf{Y},\mathbf{X}) = \frac{p(\mathbf{Y}|\mathbf{X},\boldsymbol{\beta})p(\boldsymbol{\beta}|\mathbf{X})}{p(\mathbf{Y}|\mathbf{X})} =^{(1)} \frac{p(\mathbf{Y}|\mathbf{X},\boldsymbol{\beta})p(\boldsymbol{\beta})}{p(\mathbf{Y}|\mathbf{X})}. \tag{2.9}$$

En (1) hemos utilizado que la distribución a priori es independiente de las observaciones. Al término $p(\mathbf{Y}|\mathbf{X})$ se le denomina **verosimilitud marginal**, y actúa como constante normalizadora. De nuevo, podemos aplicar el Corolario 1.1 para establecer una forma de calcularla:

$$p(\mathbf{Y}|\mathbf{X}) = \int_{\mathbb{R}^k} p(\mathbf{Y}|\mathbf{X},\boldsymbol{\beta})p(\boldsymbol{\beta})\mathrm{d}\boldsymbol{\beta}. \tag{2.10}$$

Ahora estamos en disposición de encontrar la distribución a posteriori del vector de efectos en el modelo lineal general:

Proposición 2.3. *En el modelo lineal Ecuación 2.1 con distribución a priori* $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0,\boldsymbol{\Sigma})$*, la distribución a posteriori del vector de efectos,* $\forall \mathbf{y} \in \mathbb{R}^N$ *es:*

$$\boldsymbol{\beta}|\mathbf{X},\mathbf{y} \rightsquigarrow \mathcal{N}(\frac{1}{\sigma^2}\mathbf{A}^{-1}\mathbf{X}^T\mathbf{y}, \mathbf{A}^{-1}),$$

donde $\mathbf{A} = \sigma^{-2}\mathbf{X}^T\mathbf{X} + \boldsymbol{\Sigma}^{-1}$.

Demostración. Por un lado, tenemos que:

$$p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta}) = \frac{1}{(2\pi)^{N/2}\sigma^N} \exp\left(-\frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^2}\right).$$

Por otro lado, tenemos que:

$$p(\boldsymbol{\beta}) = \frac{1}{(2\pi)^{k/2}|\boldsymbol{\Sigma}|^k} \exp\left(-\frac{\boldsymbol{\beta}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}}{2}\right).$$

Por lo tanto:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta})p(\boldsymbol{\beta}) &\propto \exp\left(-\frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^2}\right)\exp\left(-\frac{\boldsymbol{\beta}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}}{2}\right) \\ &\propto \exp\left(-\frac{1}{2}(-\frac{1}{\sigma^2}\mathbf{y}^T\mathbf{X}\boldsymbol{\beta} - \frac{1}{\sigma^2}\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y} + \frac{1}{\sigma^2}\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}\right) \\ &\propto \exp\left(-\frac{1}{2}(-\frac{1}{\sigma^2}\mathbf{y}^T\mathbf{X}\boldsymbol{\beta} - \frac{1}{\sigma^2}\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y} + \boldsymbol{\beta}^T(\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} - \boldsymbol{\Sigma}^{-1})\boldsymbol{\beta}\right). \end{aligned} \tag{2.11}$$

Intentemos buscar una expresión de la forma

$$(\boldsymbol{\beta}-\bar{\boldsymbol{\beta}})^T\boldsymbol{\Sigma}'^{-1}(\boldsymbol{\beta}-\bar{\boldsymbol{\beta}}) = \boldsymbol{\beta}^T\boldsymbol{\Sigma}'^{-1}\boldsymbol{\beta} + \bar{\boldsymbol{\beta}}^T\boldsymbol{\Sigma}'^{-1}\bar{\boldsymbol{\beta}} - \bar{\boldsymbol{\beta}}^T\boldsymbol{\Sigma}'^{-1}\boldsymbol{\beta} - \boldsymbol{\beta}^T\boldsymbol{\Sigma}'^{-1}\bar{\boldsymbol{\beta}}.$$

Podemos tomar

$$\boldsymbol{\Sigma}' = (\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} - \boldsymbol{\Sigma}^{-1})^{-1}, \tag{2.12}$$

y entonces por analogía con la Ecuación 2.11, tendríamos que:

$$-\bar{\beta}^T \Sigma'^{-1} \beta = -\frac{1}{\sigma^2} y^T \mathbf{X} \beta,$$

luego

$$\bar{\beta}^T = \frac{1}{\sigma^2} y^T \mathbf{X} \Sigma',$$

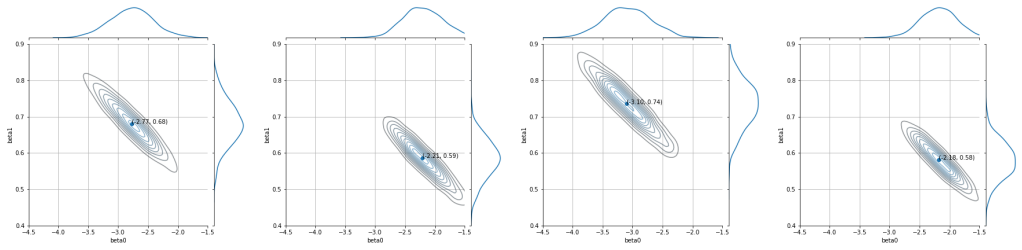
y como Σ' es una matriz simétrica, concluímos:

$$\bar{\beta} = \frac{1}{\sigma^2} \Sigma' \mathbf{X}^T \mathbf{y}. \tag{2.13}$$

Luego por lo desarrollado en la Ecuación 2.11, la distribución a posteriori es una normal con media la calculada en la Ecuación 2.13 y matriz de covarianzas la calculada en la Ecuación 2.12. □

Observación 2.1. El resultado que hemos obtenido no carece de interés: no solo se ha calculado la distribución a posteriori del modelo lineal, sino que además hemos demostrado que la distribución a priori elegida, una distribución normal, es conjugada.

Ejemplo 2.3. A partir de las distribuciones a priori expuestas en la Figura 2.2, podemos ahora calcular para la regresión llevada a cabo en la Figura 2.1 las distribuciones a posteriori. Los resultados son los correspondientes a la Figura 2.3. Se señala en ellos la media de la distribución. Al tratarse de una normal, representan el estimador MAP obtenido para la regresión lineal simple de la Figura 2.1.



(a) $\sigma_0^2 = 1, \sigma_1^2 = 1$ (b) $\sigma_0^2 = 3, \sigma_1^2 = 0.1$ (c) $\sigma_0^2 = 0.1, \sigma_1^2 = 3$ (d) $\sigma_0^2 = 3, \sigma_1^2 = 3$

Figura 2.3.: Distribuciones a posteriori correspondientes a las distribuciones a priori de la Figura 2.2 respecto de la regresión realizada en Figura 2.4. Están representadas por mapas de contorno de la función de densidad, con función de densidad de las marginales al margen. Se señala en cada panel la media obtenida en cada caso, que por tratarse de distribuciones normales, coincide con la moda y por tanto con la estimación MAP de la regresión.

Corolario 2.1. *La estimación máximo a posteriori del vector de efectos β es:*

$$\hat{\beta} = \frac{1}{\sigma^2} \left(\frac{1}{\sigma^2} \mathbf{X}^T \mathbf{X} - \Sigma^{-1} \right)^{-1} \mathbf{X}^T \mathbf{y}.$$

Demostración. Basta darse cuenta que la distribución a posteriori es una distribución normal, como se indica en la Proposición 2.3. Al ser una distribución normal, su media coincide con su moda, que es la estimación enunciada. □

Vale la pena detenerse a estudiar de nuevo la densidad predictiva. Recordamos de la Def. 1.7 que:

$$p(\mathbf{x}^*|\mathbf{X}) = \int_\Theta p(\mathbf{x}^*|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{X})\mathrm{d}\boldsymbol{\theta}.$$

Tomemos entonces un nuevo dato, $\mathbf{x}^*$, del que obtendremos la predicción $\mathbf{y}^*$. En el contexto, de la Ecuación 2.3 se tiene que $\mathbf{y}|\mathbf{X}, \boldsymbol{\beta} \rightsquigarrow \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I}_n)$ y por la Proposición 2.3 $\boldsymbol{\beta}|\mathbf{X}, y \rightsquigarrow \mathcal{N}(\sigma^{-2}\mathbf{A}^{-1}\mathbf{X}^T\mathbf{y}, \mathbf{A}^{-1})$. Así, como podemos ver desarrollado con detalle en [Bro13], obtenemos que la densidad predictiva viene determinada según

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \int_{\boldsymbol{\beta}} p(\mathbf{y}^*|\mathbf{x}^*, \boldsymbol{\beta})p(\boldsymbol{\beta}|\mathbf{X}, \mathbf{y})\mathrm{d}\boldsymbol{\beta}$$

$$\Rightarrow \mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{y} \rightsquigarrow \mathcal{N}(\sigma^{-2}\mathbf{x}^*\mathbf{A}^{-1}\mathbf{X}^T\mathbf{y}, \mathbf{x}^*\mathbf{A}^{-1}\mathbf{x}^{*T}). \quad (2.14)$$

Podemos ver que de nuevo, obtenemos una distribución normal. En la media encontramos la media de la distribución a posteriori multiplicada por los nuevos datos añadidos x^*. En la varianza encontramos una forma cuadrática de los nuevos datos con la matriz de covarianzas de la distribución a posteriori.

2.2.1. Ejemplo: Comparación de modelos

Dados unos datos o rasgos $\mathbf{X} = (X_1, \ldots, X_N)$ se obtienen las observaciones $\mathbf{Y} = (Y_1, \ldots, Y_N)$, de manera que las observaciones dependen linealmente de los datos: $\mathbf{Y} = \beta_0 + \beta_1\mathbf{X} + \boldsymbol{\varepsilon}$, con $\beta_0, \beta_1 \in \mathbb{R}$. La expresión explícita del modelo de regresión lineal simple es de la forma:

$$\begin{pmatrix} Y_1 \\ \vdots \\ Y_N \end{pmatrix} = \begin{pmatrix} 1 & X_1 \\ \vdots & \vdots \\ 1 & X_N \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{pmatrix}. \quad (2.15)$$

Recordamos las estimaciones que obteníamos mediante máxima verosimilitud y máximo a posteriori:

$$\hat{\boldsymbol{\beta}}^{\mathrm{EMV}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}, \quad (2.16)$$

$$\hat{\sigma}^{2\mathrm{EMV}} = \frac{\left\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{EMV}}\right\|^2}{N}, \quad (2.17)$$

$$\hat{\boldsymbol{\beta}}^{\mathrm{MAP}} = \frac{1}{\hat{\sigma}^{2\mathrm{EMV}}}\left(\frac{1}{\hat{\sigma}^{2\mathrm{EMV}}}\mathbf{X}^T\mathbf{X} - \boldsymbol{\Sigma}^{-1}\right)^{-1}\mathbf{X}^T\mathbf{y}. \quad (2.18)$$

Ejemplo 2.4. Utilizando los estimadores MAP calculados en la Figura 2.3, numerados respectivamente, vemos en la Figura 2.4 los resultados obtenidos. Vemos que las regresiones obtenidas basculan. La elección de una de las cinco regresiones desarrolladas dependerá de lo esperado en nuevas observaciones.

Podemos ver los efectos de las distribuciones a priori utilizadas en los resultados obtenidos. Para la MAP1, MAP2 y MAP4 (Figura 2.2, panel (a), (b) y (d)) hemos obtenido funciones con

una mayor pendiente. Esto se debe a que hemos tomado una distribución a priori con una mayor varianza en la distribución marginal de β_0, esto es, la pendiente. Sin embargo, vemos que al trabajar con una distribución a priori como la del panel (c) de la Figura 2.2, esto es, con una varianza menor en el parámetro de la pendiente, obtenemos la regresión MAP3. Vemos que este resultado es muy próximo al obtenido en la regresión EMV.

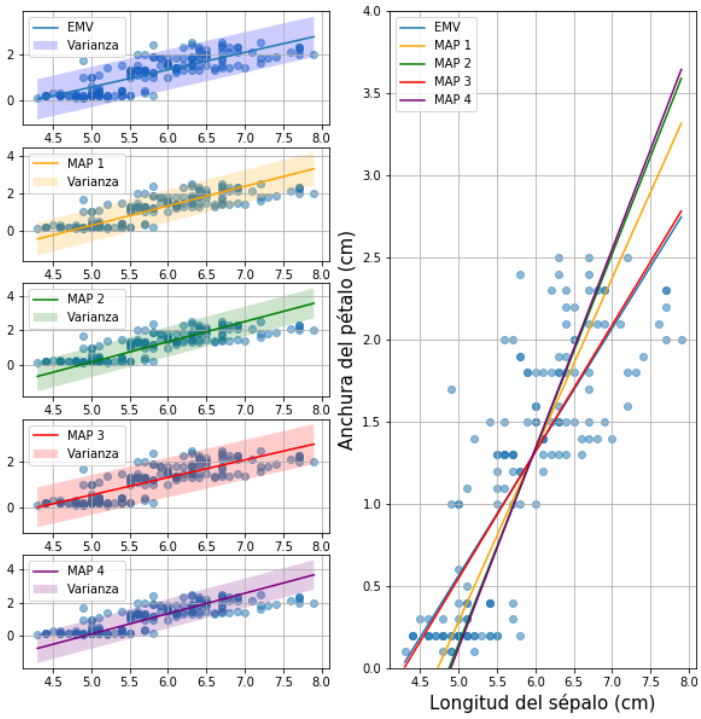


Figura 2.4.: Regresión lineal según las distribuciones a priori expresadas en la Figura 2.2 del mismo conjunto de datos tratado en la Figura 2.1

2.2.2. Funciones base

Frecuentemente, la regresión lineal carece de la expresividad necesaria para representar nuestras observaciones. Es por ello que resulta práctico proyectar los datos en un espacio de mayor dimensión para realizar allí la regresión. Por ejemplo, dado un escalar $\mathbf{x}$, podemos proyectar el dato en $\mathbb{R}^n$ al subespacio $\{(1, x, x^2, x^3, \ldots, x^{n-1}) : x \in \mathbb{R}\}$. Así, hacer regresión lineal en ese espacio es lo mismo que realizar una regresión polinómica, enriqueciendo el resultado.

Definición 2.3. Según el modelo lineal general $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, donde cada fila de $\mathbf{X} \in \mathcal{M}_{N\times k}$

corresponde a un dato, llamamos **funciones base** a funciones del tipo:

$$\begin{aligned} \phi : \mathbb{R}^k &\longrightarrow \mathbb{R}^D \\ \mathbf{x} &\longmapsto \phi(\mathbf{x}), \end{aligned} \tag{2.19}$$

donde $D > k$.

Por claridad notaremos $\mathbf{\Phi} = \phi(\mathbf{X})$ a la matriz resultado de aplicar ϕ a cada fila de $\mathbf{X}$. El modelo no deja de ser lineal, puesto que las funciones base son fijas. Ahora, sin embargo, pasa a ser de la forma

$$\mathbf{Y} = \phi(\mathbf{X})\boldsymbol{\beta} + \boldsymbol{\varepsilon} = \mathbf{\Phi}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{2.20}$$

y el vector de efectos es ahora un vector columna de $\mathbb{R}^D$. Los resultados obtenidos en ningún caso varían al efectuar este cambio, salvo notación: donde antes encontrábamos $\mathbf{X}$, ahora encontramos $\mathbf{\Phi}$.

Observación 2.2. Esto que puede parecer una novedad, no resultó nada extraño al desarrollar el modelo de regresión lineal simple: en realidad, estábamos proyectando datos escalares en el espacio $\{(1, x) : x \in \mathbb{R}\}$.

Por conveniencia, haremos unos cálculos previos a la siguiente sección, que ayudarán a motivarla. Para ello, tomaremos los resultados desarrollados en la Proposición 2.3 y en el Corolario 2.1, y los aplicaremos al nuevo modelo descrito por la Ecuación 2.20. Notaremos por sencillez, de manera análoga a como indicamos anteriormente, $\mathbf{\Phi}^* = \phi(\mathbf{x}^*)$ al resultado de aplicar ϕ por filas a los nuevos datos a partir de los que haremos predicciones:

$$\boldsymbol{\beta}|\mathbf{\Phi}, \mathbf{y} \rightsquigarrow \mathcal{N}(\frac{1}{\sigma^2}\mathbf{A}^{-1}\mathbf{\Phi}^T\mathbf{y}, \mathbf{A}^{-1}), \tag{2.21}$$

$$\hat{\boldsymbol{\beta}} = \frac{1}{\sigma^2}(\frac{1}{\sigma^2}\mathbf{\Phi}^T\mathbf{\Phi} - \mathbf{\Sigma}^{-1})^{-1}\mathbf{\Phi}^T\mathbf{y}, \tag{2.22}$$

$$\mathbf{y}^*|\mathbf{\Phi}^*, \mathbf{\Phi}, \mathbf{y} \rightsquigarrow \mathcal{N}(\sigma^{-2}\mathbf{\Phi}^*\mathbf{A}^{-1}\mathbf{\Phi}^T\mathbf{y}, \mathbf{\Phi}^*\mathbf{A}^{-1}\mathbf{\Phi}^{*T}). \tag{2.23}$$

donde $\mathbf{A} = \sigma_n^{-2}\mathbf{\Phi}^T\mathbf{\Phi} + \mathbf{\Sigma}^{-1}$.

Ejemplo 2.5. Podemos ver un ejemplo de los resultados obtenidos para una regresión polinómica de grado dos en la Figura 2.5. Para llevarla a cabo, hemos utilizado las mismas distribuciones a priori expuestas en la Figura 2.2, añadiendo un tercer parámetro al vector de efectos de varianza a priori $\sigma^2 = 1$.

De nuevo, como exponíamos para la Figura 2.4, la elección de uno u otro modelo depende del conocimiento de los datos: si se espera que los datos tengan una tendencia creciente, la regresión MAP4 o MAP2 podría ser útil para el cálculo de predicciones. En caso contrario, convendría utilizar cualquiera de las otras, y podríamos elegir una u otra dependiendo del ritmo de decrecimiento esperado.

Volviendo a lo obtenido en las ecuaciones 2.21, 2.22 y 2.23, nos enfrentamos ahora al problema de invertir la matriz $\mathbf{A} \in \mathcal{M}_D$. Nuestro objetivo es tomar D lo suficientemente grande para obtener un resultado más enriquecido. Intentemos por tanto, expresar las distribuciones de una manera más sencilla: en primer lugar, tomamos $\mathbf{K} = \mathbf{\Phi}^T\mathbf{\Sigma}\mathbf{\Phi}$. Entonces:

$$\sigma^{-2}\mathbf{\Phi}(\mathbf{K} + \sigma^2\mathbf{I}_N) = \sigma^{-2}\mathbf{\Phi}(\mathbf{\Phi}^T\mathbf{\Sigma}\mathbf{\Phi} + \sigma^2\mathbf{I}_N) = \mathbf{A}\mathbf{\Sigma}\mathbf{\Phi}.$$

Luego $\sigma^{-2}\mathbf{\Phi}^T(\mathbf{K} + \sigma^2\mathbf{I}_N) = \mathbf{A}\mathbf{\Sigma}\mathbf{\Phi}^T$ y por tanto $\sigma^{-2}\mathbf{A}^{-1}\mathbf{\Phi}^T = \mathbf{\Sigma}\mathbf{\Phi}^T(\mathbf{K} + \sigma^2\mathbf{I}_N)^{-1}$.

Así, hemos resuelto el problema en la media, donde hemos pasado de tener que invertir $\mathbf{A} \in \mathcal{M}_D$, a tener que invertir $(\mathbf{K} + \sigma^2 \mathbf{I}_N) \in \mathcal{M}_N$. Para la varianza, utilizaremos el siguiente lema, demostrado en [RW06]:

Lema 2.1. *(Fórmula de Woodbury, Sherman y Morrison) Sean* $\mathbf{P} \in \mathcal{M}_n, \mathbf{Q} \in \mathcal{M}_m, \mathbf{U}, \mathbf{V} \in \mathcal{M}_{m \times n}$*, entonces, de existir:*

$$(\mathbf{P} + \mathbf{U}^T \mathbf{Q} \mathbf{V})^{-1} = \mathbf{P}^{-1} - \mathbf{P}^{-1} \mathbf{U}^T (\mathbf{Q}^{-1} + \mathbf{V} \mathbf{P}^{-1} \mathbf{U}^T)^{-1} \mathbf{V} \mathbf{P}^{-1}. \tag{2.24}$$

Por lo tanto, podemos invertir $\mathbf{A}$ tomando $\mathbf{U} = \mathbf{V} = \mathbf{\Phi}, \mathbf{P} = \mathbf{\Sigma}^{-1}, \mathbf{Q} = \mathbf{I}_N$, obteniendo:

$$\mathbf{A}^{-1} = \mathbf{\Sigma} - \mathbf{\Sigma} \mathbf{\Phi}^T (\mathbf{I}_N + \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^T)^{-1} \mathbf{\Phi} \mathbf{\Sigma}.$$

Por tanto, podemos concluir con la versión optimizada de la distribución predictiva, que sería:

$$\begin{aligned} \mathcal{N}(\sigma^{-2} \mathbf{\Phi}^* (\mathbf{\Sigma} - \mathbf{\Sigma} \mathbf{\Phi}^T (\mathbf{I}_N + \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^T)^{-1} \mathbf{\Phi} \mathbf{\Sigma}) \mathbf{\Phi}^T \mathbf{y}, \mathbf{\Phi}^* (\mathbf{\Sigma} - \mathbf{\Sigma} \mathbf{\Phi}^T (\mathbf{I}_N + \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^T)^{-1} \mathbf{\Phi} \mathbf{\Sigma}) \mathbf{\Phi}^{*T}) \\ = \mathcal{N}(\mathbf{\Phi}^* \mathbf{\Sigma} \mathbf{\Phi}^T (K + \sigma^2 \mathbf{I}_N)^{-1} \mathbf{y}, \\ \mathbf{\Phi}^* \mathbf{\Sigma} \mathbf{\Phi}^{*T} - \mathbf{\Phi}^* \mathbf{\Sigma} \mathbf{\Phi}^T (K + \sigma^2 \mathbf{I}_N)^{-1} \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^{*T}). \end{aligned} \tag{2.25}$$

Vemos que en cualquier caso, los datos siempre intervienen en esta distribución según:

$$\mathbf{\Phi}^* \mathbf{\Sigma} \mathbf{\Phi}^T, \qquad \mathbf{\Phi}^* \mathbf{\Sigma} \mathbf{\Phi}^{*T}, \qquad \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^{*T}, \qquad \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Phi}^T.$$

Podemos considerar por tanto que dado cualquier par de datos como vectores fila, añadidos o no, $\mathbf{x}, \mathbf{x}'$, entonces estos intervienen siempre según $\phi(\mathbf{x}) \mathbf{\Sigma} \phi(\mathbf{x}')^T$. Por conveniencia para la siguiente sección, haremos la siguiente definición:

Definición 2.4. Dada una matriz de covarianzas $\mathbf{\Sigma}$, definimos su **función de covarianza** o ***kernel*** como:

$$\begin{aligned} k : \mathbb{R}^N \times \mathbb{R}^N &\longrightarrow \mathbb{R} \\ (\mathbf{x}, \mathbf{x}') &\longmapsto k(\mathbf{x}, \mathbf{x}') = \mathbf{x} \mathbf{\Sigma} \mathbf{x}'^T. \end{aligned} \tag{2.26}$$

2.3. Modelo lineal general mediante procesos gaussianos

A partir de los conceptos introducidos en la sección anterior, se desarrollará ahora la definición de proceso gaussiano, primero desde cero y luego como distribución sobre funciones, a partir de las funciones base de la regresión bayesiana. Esto nos permitirá afrontar la regresión desde un nuevo punto de vista, que nos permite calcular predicciones a partir de una mayor abundancia de información.

Definición 2.5. Llamamos **proceso gaussiano** a un conjunto de variables aleatorias tal que cada subconjunto finito describe una distribución normal mutivariante. Queda definido a partir de una función media $m(\cdot)$ y una función covarianza o kernel $k(\cdot, \cdot)$. Lo notamos $f \rightsquigarrow GP(m(\cdot), k(\cdot, \cdot))$. Así, $f = \{f(x) : x \in \mathcal{X}\}$ donde $\mathcal{X}$ es el conjunto ordenado de índices.

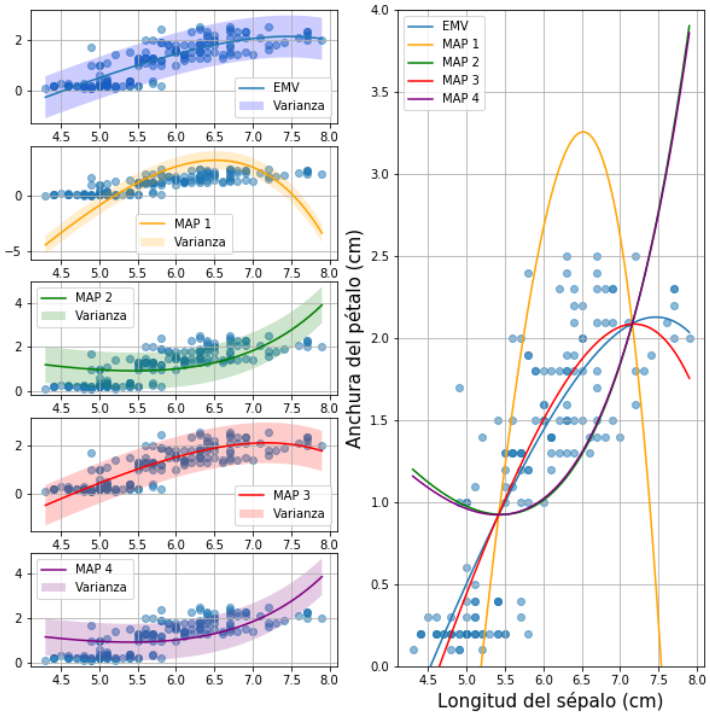


Figura 2.5.: Regresión lineal según las distribuciones a priori expresadas en la Figura 2.2 del mismo conjunto de datos tratado en la Figura 2.1 y Figura 2.4. En este caso, hemos utilizado como función base $\phi(\mathbf{X}) = (1, \mathbf{X}, \mathbf{X}^2)$.

Así, tomando un subconjunto finito $x_1, \ldots, x_n$ obtenemos:

$$\begin{aligned}(f(x_1), \ldots, f(x_n)) &\rightsquigarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \text{donde} \\ \boldsymbol{\mu} &= m(x_1, \ldots, x_n), \\ \boldsymbol{\Sigma} &= k((x_1, \ldots, x_n), (x_1, \ldots, x_n)).\end{aligned} \tag{2.27}$$

El estudio de los procesos gaussianos estará centrado en conjuntos $\mathcal{X}$ infinitos, ya que el caso finito no es más que estudiar una distribución gaussiana mutivariante al uso. Generalmente, el conjunto de índices suele definirse como una variable temporal. Sin embargo, aquí se considerará el conjunto de posibles nuevos datos, siendo el más general $\mathcal{X} = \mathbb{R}^N$.

Observación 2.3.

- $m(x) = \mathbb{E}[f(x)]$,
- $k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))]$.

Para aportar claridad, y sin pérdida de generalidad, podemos tomar $m(x) = 0$

Ya hemos desarrollado un ejemplo de proceso gaussiano: el modelo de regresión bayesiano. En él, tomando $\mathbf{y} = f(\mathbf{X}) = \phi(\mathbf{X})\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ con una distribución a priori $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0, \boldsymbol{\Sigma})$

$$\begin{aligned}\mathbb{E}[f(\mathbf{X})] &= \phi(\mathbf{X})\mathbb{E}[\boldsymbol{\beta}] + \mathbb{E}[\boldsymbol{\varepsilon}] = 0,\\ \mathbb{E}[f(\mathbf{X})f(\mathbf{X}')] &= \phi(\mathbf{X})\boldsymbol{\Sigma}\phi(\mathbf{X}'),\end{aligned}$$

tal y como definíamos en la Def. 2.4. Veamos a continuación cómo utilizar los procesos gaussianos para obtener una generalización de la regresión lineal bayesiana a una regresión lineal en la que se utilicen infinitas funciones base.

2.3.1. Función de base radial

Consideremos el problema de regresión lineal bayesiana $f(\mathbf{X}) = \phi(\mathbf{X})\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, donde $\phi = (\phi_{c_1}, \ldots, \phi_{c_N})$ con $c_1, \ldots, c_N \in \mathbb{R}$:

$$\begin{aligned}\phi_c : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto \exp\left(-\frac{(x-c)^2}{2\ell^2}\right).\end{aligned} \tag{2.28}$$

Llamaremos al parámetro $\ell \in \mathbb{R}^+$ **longitud de escala característica**. Tomemos el caso concreto en que la distribución a priori es $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0, \sigma^2\mathbf{I}_N)$. Entonces, por la Def. 2.4:

$$k(x, x') = \sigma^2 \sum_{i=1}^{N} \phi_{c_i}(x)\phi_{c_i}(x').$$

Ahora, consideremos el caso en que $c \in (a, b) \subset \mathbb{R}$, esto es, el caso en que tenemos funciones base, centradas uniformemente en los puntos del intervalo (a, b), y modificando la distribución a priori, escalando la varianza según el número de funciones base seleccionadas, $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0, \frac{\sigma^2}{N}\mathbf{I}_N)$. Tomando límites en el número de funciones base:

$$\lim_{N\to\infty} \frac{\sigma^2}{N} \sum_{i=1}^{N} \phi_{c_i}(x)\phi_{c_i}(x') =^{(1)} \sigma^2 \int_a^b \phi_c(x)\phi_c(x')\mathrm{d}c.$$

En (1) hemos utilizado que el límite que estábamos calculando era una suma de Riemann. Ahora, podemos hacer tender los límites de la integral a infinito y calcularla. Podemos ver el resultado con más detalle en [RW06]:

$$\begin{aligned}k(x, x') = \sigma^2 \int_{-\infty}^{\infty} \phi_c(x)\phi_c(x')\mathrm{d}c = \\ \sigma^2 \int_{-\infty}^{\infty} \exp\left(-\frac{(x-c)^2}{2\ell^2}\right) \exp\left(-\frac{(x'-c)^2}{2\ell^2}\right) \mathrm{d}c = \\ \sqrt{\pi}\ell\sigma^2 \exp\left(-\frac{(x-x')^2}{2(\sqrt{2}\ell)^2}\right).\end{aligned} \tag{2.29}$$

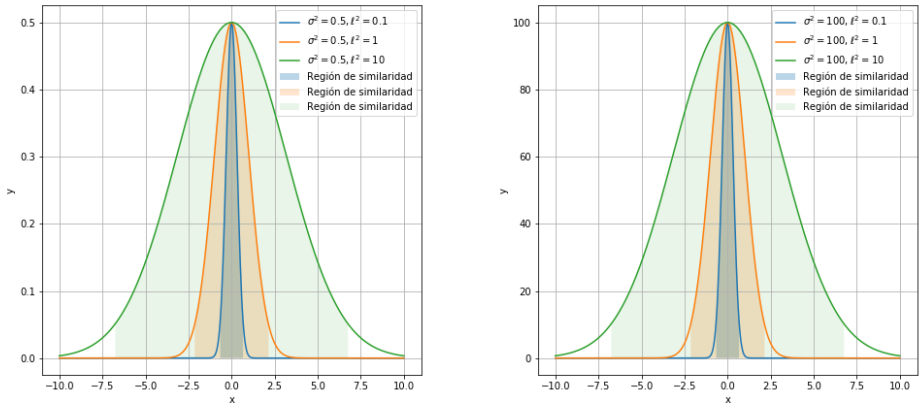
Pese a que hemos llevado a cabo el caso escalar, la generalización al caso vectorial es análoga.

Definición 2.6. Dado un proceso gaussiano $GP(0, k(\cdot,\cdot))$, decimos que el kernel es del tipo **función de base radial** o ***RBF*** si es de la forma:

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right). \tag{2.30}$$

Así, podemos ver un proceso gaussiano como una distribución sobre un espacio de funciones. En concreto, cuando el kernel es del tipo *RBF*, el proceso gaussiano correspondiente puede considerarse como una distribución sobre las funciones del tipo 2.28.

En la Figura 2.6 podemos ver el efecto que tiene sobre los resultados la determinación a priori de distintos valores de los hiperparámetros en la RBF. Se señala en el área sombreada aquellos valores en los que se puede considerar que hay dependencia o similaridad con el 0, valor en el que se ha fijado la RBF. Dado que dicha función describe una covarianza, cuanto más cercano a 0 esté su valor entre $\mathbf{x}$ y $\mathbf{x}'$, esto es, cuánto más cercano a cero esté $K(\mathbf{x}, \mathbf{x}')$, más incorrelados estarán dichos datos. Por tanto, mientras que σ afecta a la amplitud de los datos, como se observa en el eje Y, la longitud de escala determina la proximidad con la que un dato se considera similar o correlado con otro.



(a) RBF con $\sigma^2 = 0.05$, variando ℓ^2.

(b) RBF con $\sigma^2 = 100$, variando ℓ^2.

Figura 2.6.: Gráfica de la función de base radial $K(x, 0) = \sigma^2 \exp(-x^2/2\ell^2)$ para distintos valores de los hiperparámetros σ^2 y ℓ^2. Se puede apreciar como σ^2 afecta a la amplitud de los datos en el eje Y, mientras que la longitud de escala característica ℓ^2 determina la proximidad de los datos que son similares. El área sobreada representa aquellos valores para los que la RBF se encuentra por encima del 10 % del valor de σ.

2.3.1.1. Selección de los hiperparámetros.

El problema de selección de los hiperparámetros se afronta desde el punto de vista de la inferencia bayesiana. Podemos identificar dos niveles de inferencia en el desarrollo de, por ejemplo, un problema de regresión. El nivel más bajo trata de inferir los parámetros

del modelo, esto es, el vector de efectos, $\boldsymbol{\beta}$ y la desviación o ruido $\boldsymbol{\varepsilon}$. En el segundo nivel, encontramos la determinación de los hiperparámetros, que en el caso de la RBF son σ^2 y ℓ. Incluso se puede considerar un tercer nivel más alto de inferencia, la selección del modelo, esto es, la determinación de la estructura del problema. Este último problema escapa al objetivo de este trabajo y puede verse con detalle en [RW06].

Podemos afrontar la inferencia de los hiperparámetros desde el enfoque bayesiano, mediante el teorema de Bayes. En el primer nivel, teníamos la distribución a posteriori del vector de efectos $\boldsymbol{\beta}$:

$$p(\boldsymbol{\beta}|\mathbf{y},\mathbf{X},\sigma,\ell) = \frac{p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta})p(\boldsymbol{\beta}|\sigma,\ell)}{p(\mathbf{y}|\mathbf{X},\sigma,\ell)}, \tag{2.31}$$

donde $p(\mathbf{y}|\mathbf{X},\boldsymbol{\beta})$ es la verosimilitud, $p(\boldsymbol{\beta}|\sigma,\ell)$ es la distribución a priori del vector de efectos y $p(\mathbf{y}|\mathbf{X},\sigma,\ell)$ es la evidencia o verosimilitud marginal. Siempre se trata de elegir una distribución a priori acorde con la información de la que se disponga, haciéndola lo más general posible en caso de que dicha información no sea abundante.

De manera análoga, podemos determinar para el segundo nivel de inferencia una distribución a posteriori sobre los hiperparámetros:

$$p(\sigma,\ell|\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{y}|\mathbf{X},\sigma,\ell)p(\sigma,\ell)}{p(\mathbf{y}|\mathbf{X})}, \tag{2.32}$$

donde vemos que la evidencia hace el papel de la verosimilutd de los hiperparámetros. Encontramos la distribución a priori de los hiperparámetros también, $p(\sigma,\ell)$. En este caso, la constante normalizadora, esto es, la verosimilitud marginal de los hiperparámetros, es:

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{X},\sigma,\ell)p(\sigma,\ell)\mathrm{d}\sigma\mathrm{d}\ell, \tag{2.33}$$

que podemos interpretar como la esperanza de todos las posibles evidencias, cuando variamos los hiperparámetros. La evaluación de esta integral suele ser difícil, por lo que en la práctica no se suele hacer inferencia bayesiana en este nivel de inferencia. En su lugar, se suele hacer inferencia clásica, en concreto estimaciones máximo verosímiles de los hiperparámetros. Esto recibe el nombre de la estimación **II máximo verosímil** (ML-II). En caso de querer afrontar este problema desde el enfoque bayesiano, puede tratarse de aproximar la integral de la Ecuación 2.33 mediante métodos numéricos.

2.3.2. Densidad predictiva

Como anteriormente, lo interesante de la regresión lineal reside en obtener predicciones a partir de nuevos datos. Para ello, vamos a centrar nuestro estudio en el cálculo explícito de las distribuciones predictivas de la distribución de $f(\mathbf{X})$ vista en la Def. 2.5. De ahora en adelante, notaremos por claridad $\mathbf{f} = f(\mathbf{X})$ y $\mathbf{f}^* = f(\mathbf{X}^*)$, siendo $\mathbf{X}$ la matriz de diseño y $\mathbf{X}^*$ los nuevos datos que se añaden para llevar a cabo la predicción $\mathbf{f}^*$.

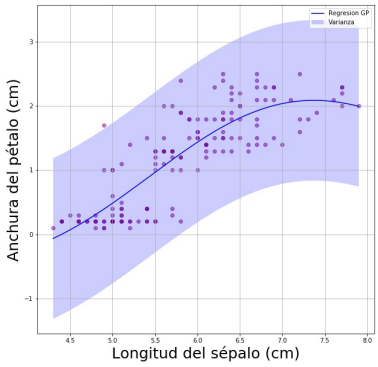


Figura 2.7.: Regresión mediante procesos gaussianos para los datos descritos en el E.j. 2.1. Los hiperparámetros óptimos obtenidos son $\sigma = 1.36$ y una longitud de escala de 2.13. Por lo tanto, la distribución a priori es $\mathrm{GP}(0, 1.36^2 \exp(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2.13^2))$.

2.3.2.1. Predicciones a partir de datos exactos.

En primer lugar, centrémonos en el caso en que conocemos exactamente el valor de f para cada $\mathbf{X}$. En ese caso, dado un nuevo dato $\mathbf{X}^*$, tenemos que:

$$(\mathbf{f}, \mathbf{f}^*) \rightsquigarrow \mathcal{N}(0, \begin{pmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{pmatrix}. \tag{2.34}$$

Para N filas de la matriz de diseño, y N^* nuevos datos, se tiene que $K(\mathbf{X}^*, \mathbf{X})$ es la matriz $N \times N^*$ de covarianzas de $\mathbf{X}^*$ y $\mathbf{X}$. Dado que la distribución de la normal mutivariante es conocida, es fácil obtener la distribución predictiva, esto es, la distribución a posteriori de los nuevos datos:

$$\mathbf{f}^* | \mathbf{X}^*, \mathbf{X}, \mathbf{f} \rightsquigarrow \mathcal{N}(K(\mathbf{X}^*, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}\mathbf{f}, \\ K(\mathbf{X}^*, \mathbf{X}^*) - K(\mathbf{X}^*, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}K(\mathbf{X}, \mathbf{X}^*)). \tag{2.35}$$

2.3.2.2. Predicciones a partir de datos con ruido.

Volvamos al modelo de Gauss-Markov con el que comenzamos en la Subsección 2.1.1, esto es: $\mathbf{y} = f(\mathbf{X}) + \boldsymbol{\varepsilon}$ con $f(\mathbf{X}) \rightsquigarrow GP(0, K(\cdot, \cdot))$. De aquí, deducimos de manera inmediata:

$$\mathbb{E}[\mathbf{y}\mathbf{y}'^T] = K(\mathbf{X}, \mathbf{X}') + \sigma^2 \delta(\mathbf{X}, \mathbf{X}'), \quad \text{i.e.} \quad E[\mathbf{y}\mathbf{y}^T] = K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}_N, \tag{2.36}$$

donde δ se refiere a la delta de Kronecker. Al añadir ruido, lo que obtenemos es la misma Ecuación 2.34 modificada por la matriz diagonal de varianzas del ruido:

$$(\mathbf{y}, \mathbf{f}^*) \rightsquigarrow \mathcal{N}(0, \begin{pmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}_N & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{pmatrix}. \tag{2.37}$$

De nuevo, basta aplicar las propiedades conocidas de la distribución normal mutivariante para obtener el resultado análogo a la Ecuación 2.35:

$$\mathbf{f}^*|\mathbf{X},\mathbf{y},\mathbf{X}^* \rightsquigarrow \mathcal{N}(\bar{\mathbf{f}}^*, \mathrm{Cov}(\mathbf{f}^*)), \tag{2.38}$$

$$\bar{\mathbf{f}}^* = K(\mathbf{X}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I}_N)^{-1}y, \tag{2.39}$$

$$\mathrm{Cov}(\mathbf{f}^*) = K(\mathbf{X}^*,\mathbf{X}^*) - K(\mathbf{X}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I}_N)^{-1}K(\mathbf{X},\mathbf{X}^*). \tag{2.40}$$

Observación 2.4. Lo que hemos obtenido en la Ecuación 2.39 y en la Ecuación 2.40 es perfectamente análogo a lo obtenido en la Ecuación 2.25, identificando $K(\mathbf{U},\mathbf{V}) = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ donde $\mathbf{U}$ y $\mathbf{V}$ se corresponden según el caso con $\mathbf{X}$ o $\mathbf{X}^*$.

Una de las propiedades por las que los procesos gaussianos son de extendido uso puede observarse en la Ecuación 2.38: las observaciones de $\mathbf{y}$ dado $\mathbf{X}$ no juegan ningún papel en la varianza a la hora de hacer predicciones.

Además vale la pena detenerse a analizar la interpretación de la Ecuación 2.40: por un lado, $K(\mathbf{X}^*,\mathbf{X}^*)$ no es más que la covarianza según la distribución a priori, restado un término, $K(\mathbf{X}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I}_N)^{-1}K(\mathbf{X},\mathbf{X}^*)$. Este término es positivo, y forma parte de la información que reside en la varianza de las observaciones. Esto está de nuevo en la línea expuesta en la Sección 1.2: recoger la información a priori y de las observaciones en la distribución a posteriori, y en última instancia, en la distribución predictiva.

Ejemplo 2.6. Para los datos expuestos en el E.j. 2.1, llevamos a cabo la regresión lineal mediante procesos gaussianos, utilizando el ruido calculado mediante EMV en dicho ejemplo. Para ello, optimizamos en primer lugar los hiperparámetros, de manera que obtenemos $\ell = 2.13$ y $\sigma = 1.36$. Por lo tanto, la distribución a priori es un proceso gaussiano con media 0 y función *kernel* $K(\mathbf{x},\mathbf{x}') = 1.36^2 \exp(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2.13^2})$. El resultado de la regresión es como se expone en la Figura 2.7. Vemos que se ajusta mucho mejor a los datos que las regresiones de la Figura 2.4 y la Figura 2.5. Esto puede llevar, no obstante, a un sobreajuste a la hora de llevar a cabo predicciones.

2.4. Clasificación mediante modelos lineales

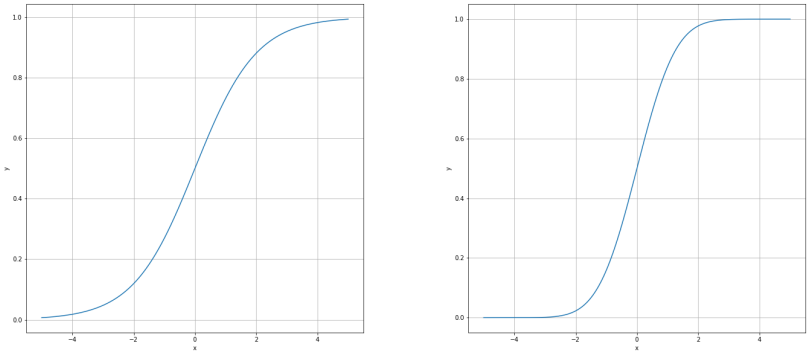
La facilidad con la que se llevan a cabo los cálculos para los modelos lineales reside casi por completo en que el modelo de observación sigue una distribución normal. Cuando esto no ocurre, el problema pierde tratabilidad. Es por ello que utilizaremos unas funciones, las funciones sigmoides, que tratan de resolver el problema y convertir verosimilitudes intratables en verosimilitudes que sigan distribuciones gaussianas.

Definición 2.7. Llamamos **sigmoide** a una función $\mathrm{sig} \in \mathcal{C}[0,1]$ estrictamente creciente:

$$\begin{aligned} \mathrm{sig} : \mathbb{R}^D &\longrightarrow [0,1] \\ x &\longmapsto \mathrm{sig}(x). \end{aligned} \tag{2.41}$$

Este tipo de funciones de observación nos los encontraremos sobre todo en problemas de clasificación. En esta sección, abordaremos el problema de la clasificación binaria. En los problemas de este tipo, notaremos:

- El conjunto de datos observados o rasgos, $X = \{\mathbf{x}_i \in R^M, i = 1,\dots,N\}$.



(a) $y = (1 + \exp(-x))^{-1}$ (b) $y = P(\mathcal{N}(0,1) \leq x)$

Figura 2.8.: Funciones sigmoides más extendidas.

- Para cada vector de rasgos $\mathbf{x}_i$, se cuenta con una etiqueta o anotación, esto es, se establece la clase a la que pertenece. Esta etiqueta se notará y_i, siendo Y el conjunto de todas las etiquetas. Siguiendo la literatura, en el caso de clasificación binaria, se toma el conjunto $Y = \{1, -1\}$ para nombrar a las dos clases.
- El conjunto de todos los datos $\mathcal{D} = \{(y_i, \mathbf{x}_i) : y_i \in -1, 1, i = 1, \ldots, N\}$.

Decimos que un problema de clasificación es **supervisado** cuando se cuenta con todas las anotaciones de cada dato. En caso contrario, habría que inferir también las anotaciones, problema que recibe el nombre de clasificación **no supervisada**. A lo largo de esta sección, trataremos la introducción a los problemas de clasificación supervisada binaria.

Ahora, usando el teorema de Bayes (Ecuación 1.7) podemos descomponer la probabilidad conjunta de dos formas:

$$p(y, x) = p(y)p(x|y), \qquad (2.42) \qquad\qquad p(y, x) = p(x)p(y|x). \qquad (2.43)$$

Cuando utilizamos la Ecuación 2.42 decimos que el problema de clasificación es de tipo **generativo**, mientras que si utilizamos la Ecuación 2.43 diremos que es de tipo **discriminativo**. El modelo generativo trata de inferir la distribución condicionada a cada clase $p(x|y)$ para calcular, utilizando las distribuciones a priori de cada clase, las distribuciones a posteriori.

Sin embargo, el modelo discriminativo trata de inferir directamente la distribución a posteriori $p(y|x)$.

Mediante el modelo generativo, podríamos utilizar el enfoque bayesiano tal y como se ha expuesto hasta ahora, asignando unas distribuciones a priori normales a cada clase. Sin embargo, esto sería asumir demasiada información, lo que restaría credibilidad y riqueza al resultado. Así, se tratará de utilizar una función sigmoide adecuada para tratar de obtener distribuciones a posteriori normales. Cuando se utiliza la función sigmoide del panel (a) de la Figura 2.8 el problema de clasificación recibe el nombre de **regresión lineal logistica**. Cuando se utiliza la función sigmoide del panel (b), llamamos al problema de clasificación **regresión lineal *probit***.

Por cohesión con lo expuesto hasta ahora, trabajaremos con el modelo discriminativo. Esto nos permitirá, por ejemplo, utilizar las propiedades de la evidencia desarrolladas en la Subsección 1.4.2.

2.4.1. Clasificación mediante el modelo lineal general bayesiano

Utilicemos en primer lugar el modelo lineal bayesiano para desarrollar una primera idea del procedimiento que habrá que seguir al utilizar el modelo lineal mediante procesos gaussianos, que es el objetivo final de este capítulo.

Así, consideremos la distribución a priori utilizada en las anteriores secciones $\boldsymbol{\beta} \rightsquigarrow \mathcal{N}(0, \boldsymbol{\Sigma})$. Consideremos ahora un modelo exacto, esto es, en el que se conocen las anotaciones $y_i, \forall i \in \{1, \dots, N\}$, que son independientes entre sí, y por tanto de la forma:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}. \tag{2.44}$$

La verosimilitud, dada una función sigmoide sig, será:

$$p(y = +1|\mathbf{X}, \boldsymbol{\beta}) = \text{sig}(\mathbf{X}\boldsymbol{\beta}). \tag{2.45}$$

Como es el caso de clasificación binaria, esto quiere decir que

$$p(y = +1|\mathbf{X}, \boldsymbol{\beta}) + p(y = -1|\mathbf{X}, \boldsymbol{\beta}) = 1, \text{ i.e. } p(y = -1|\mathbf{X}, \boldsymbol{\beta}) = 1 - \text{sig}(\mathbf{X}\boldsymbol{\beta}). \tag{2.46}$$

Tanto en la regresión *probit* como en la logística las sigmoides son simétricas, de modo que $\text{sig}(-x) = 1 - \text{sig}(x)$. Por tanto, en estos casos y en otros similares, podemos abreviar la expresión de la verosimilitud como sigue:

$$p(y_i|\mathbf{X}, \boldsymbol{\beta}) = \text{sig}(y_i \mathbf{f}_i), \tag{2.47}$$

donde hemos utilizado la notación ya introducida en la sección anterior, $\mathbf{f}_i = f(\mathbf{X}_i) = \mathbf{X}_i\boldsymbol{\beta}$. Así, tal y como podemos encontrar en [RW06], el logaritmo de la función de densidad de la distribución a posteriori es de la forma:

$$\ln p(\boldsymbol{\beta}|\mathbf{X}, y) = -\frac{1}{2}\boldsymbol{\beta}\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}^T + \sum_{i=1}^{N} \ln \text{sig}(y_i \mathbf{f}_i) \tag{2.48}$$

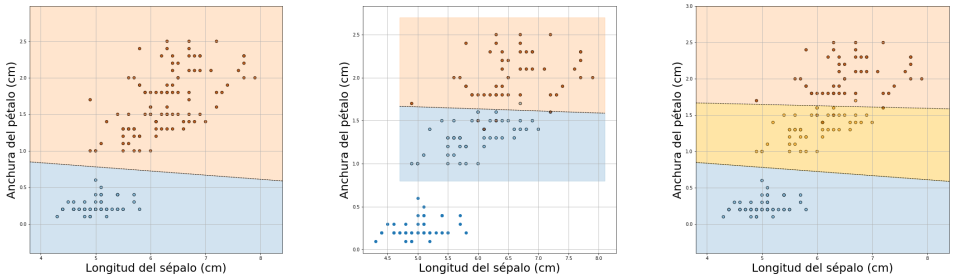
Pese a que se pierde el carácter gaussiano de la distribución a posteriori, tal y como se detalla en [RW06], para funciones sigmoides el hessiano de $p(\boldsymbol{\beta}|\mathbf{X}, y)$ es definido negativo y por tanto la Ecuación 2.48 es cóncava en $\boldsymbol{\beta}$ para un conjunto $\mathcal{D}$ fijo. Por tanto, tiene un único máximo, que será la estimación MAP, y basta con utilizar el algoritmo de Newton, por ejemplo, para encontrarlo.

Ejemplo 2.7. Podemos ver que pese a utilizar una sigmoide, no deja de ser un problema de regresión lineal. Una forma sencilla de pasar del problema de clasificación binario al de clasificación multiclase se expone en la Figura 2.9.

Los datos, como se detallaba en el E.j. 2.1, están extraídos de la base de datos iris, donde se estudian características de tres especies distintas de flores: setosa, virginica y versicolor. Para diferenciarlas, utilizaremos la regresión lineal logística binaria. En primer lugar, trataremos de distinguir la especie setosa de las demás, como vemos en el panel (a). Una vez determinada la clasificación, puede verse que la región establecida por la recta de regresión coincide totalmente con la clasificación original.

Ahora, tomamos los datos restantes de especies distintas a la setosa, y repetimos el proceso: volvemos a tener un problema de clasificación binario. El resultado de distinguir la virginica de la versicolor podemos verlo en la región sombreada del panel (b).

Finalmente, basta con dibujar las rectas de regresión obtenidas, tratando de que cada una no interfiera con la zona delimitada por la etapa anterior. En nuestro ejemplo, el resultado final se puede observar en el panel (c).



(a) Setosa diferenciada de versicolor y virginica.

(b) Versicolor diferenciada de virginica.

(c) Diferencia entre las tres.

Figura 2.9.: Clasificación mediante regresión logística para los mismos datos utilizados en el E.j. 2.1

.

Así, se establece una manera de proceder sencilla para el problema de clasificación multiclase: en primer lugar, se distingue una clase de las demás. De entre el resto, se repite el proceso, distinguiendo de nuevo la primera de ellas de las demás. De la misma manera, se continúa hasta que se distinguen la penúltima clase y la última. Al acabar, tendremos las rectas de regresión que distingue cada una de las clases entre sí.

2.4.2. Clasificación mediante procesos gaussianos

Para extender la clasificación introducida en la subsección anterior, el procedimiento es análogo al que hacíamos en la Sección 2.3: utilizamos como distribución a priori un proceso gaussiano $f(\mathbf{X})$ y lo convertimos en una variable aleatoria utilizando una función sigmoide. Así, podemos interpretar el grado de credibilidad de la evidencia como sigue:

Definición 2.8. En el contexto anterior, llamamos **probabilidad de clase** y lo notamos como $\pi = \pi(\mathbf{X})$ a lo siguiente:

$$\pi(\mathbf{X}) = p(y = +1|\mathbf{X}) = \text{sig}(f(\mathbf{X})). \tag{2.49}$$

Como vemos, la importancia reside en el conocimiento de la nueva distribución de probabilidad de clase $\pi(\mathbf{X})$, y no en el proceso gaussiano, cuyo uso tan solo es útil para desarrollar una teoría robusta para el problema de clasificación. Por ejemplo, pese a que se ha tomado intencionadamente un modelo sin ruido, se puede probar que un modelo con ruido es equivalente a otro sin ruido, salvo un cambio en la verosimilitud.

Así, el problema queda dividido en dos etapas: calcular la densidad predictiva, y para dicha predicción, calcular su probabilidad de clase, y por tanto se trata de calcular:

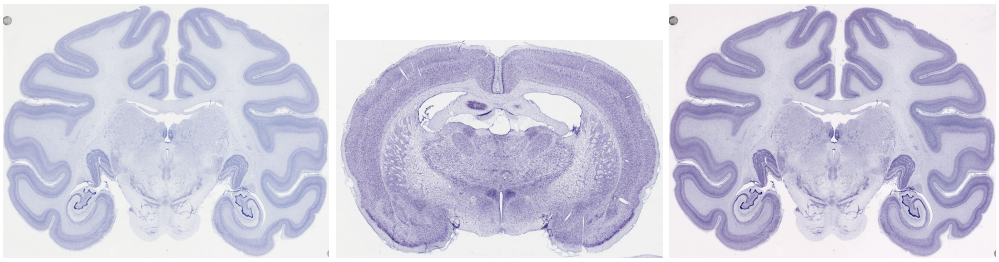
$$p(\mathbf{f}^*|\mathbf{X},\mathbf{y},\mathbf{X}^*) = \int p(\mathbf{f}^*|\mathbf{X},\mathbf{X}^*,\mathbf{f})p(\mathbf{f}|\mathbf{X},\mathbf{y})\mathrm{d}\mathbf{f}, \tag{2.50}$$

$$\pi^* = p(\mathbf{y}^* = +1|\mathbf{X},\mathbf{y},\mathbf{X}^*) = \int \mathrm{sig}(\mathbf{f}^*)p(\mathbf{f}^*|\mathbf{X},\mathbf{y},\mathbf{X}^*)\mathrm{d}\mathbf{f}^*. \tag{2.51}$$

Cuando desarrollábamos la regresión, el calculo de la correspondiente Ecuación 2.50 era sencillo debido a que todas las distribuciones eran normales, y por tanto analíticamente tratables. Así, puesto que ahora hemos perdido esa condición, tanto esta como la Ecuación 2.51 no son calculables de manera analítica. En nuestro caso (clasificación binaria), ambas ecuaciones son integrales en $\mathbb{R}$ y por tanto se pueden calcular por los métodos numéricos usuales. En el Capítulo 4 trataremos de resolver el problema para el caso de la clasificación multiclase mediante *crowdsourcing*. Este problema requiere de un estudio con más detenimiento, porque es un problema de clasificación con etiquetas con ruido.

3. Normalización de color de imágenes

En todo proceso de aprendizaje automático, como en los de regresión y clasificación desarrollados en el capítulo anterior, se requiere una fase previa de normalización de los datos. En este proceso, se trata de aplicar transformaciones que hagan los datos más tratables, sin modificar su poder de representación. Cuando se trata de imágenes, uno de los aspectos a tener en cuenta en el proceso de normalización es el color. Por ejemplo, a la hora de desarrollar un modelo de clasificación para imágenes histológicas, puede que la tinción utilizada para unas difiera de otras. Por ello, en este capítulo se expone una forma sencilla de acometer este problema, basada en el artículo [RAGS01]. Vemos un ejemplo de lo que se desarrollará en el capítulo en la Figura 3.1.



(a) Original (b) Color objetivo (c) Resultado

Figura 3.1.: Imagen del cerebro de la especie *macaca mulatta*.

3.1. Cambio de sistema de color

En el contexto de la ciencia de datos, las imágenes son un problema ampliamente abordado. Una imagen está compuesta por **píxeles.** Cada píxel almacena la información de la cantidad de colores primarios. Al dato correspondiente a cada color primario se le denomina **canal**. Por ejemplo, en una imagen en blanco y negro cada píxel almacena la información de la cantidad de negro, por lo que tiene tan solo un canal. Mientras tanto, en un sistema con colores primarios rojo, verde y azul, (denominado ***RGB***) cada píxel tiene tres canales.

El sistema *RGB* es el que encontramos en la mayoría de formatos digitales pero entraña un problema: la correlación entre los valores de los diferentes canales es muy alta. Así, si en el canal azul encontramos un valor muy alto, podemos esperar un valor muy alto también en los canales rojo y verde.

Así, podemos ver una imagen como un conjunto de datos. Estos datos se almacenan en píxeles: si la imagen tiene H píxeles de altura y W píxeles de ancho, tenemos un conjunto de $H \times W$ píxeles. En cada píxel se almacena la información de los canales. En el caso *RGB*, lo más extendido son las imágenes de 24 bits, esto es, 8 bits para cada canal. Así, cada píxel

almacena la información de la cantidad de rojo, verde y azul. Esta cantidad de color se representa mediante el brillo, cuyo valor oscila entre 0 y 255. Por tanto, podemos trabajar con una imagen como elemento de $\mathbb{R}^{H\times W\times 3}$, esto es, con tres canales en cada píxel, donde cada valor estará entre 0 y 255.

Para solucionar este problema, podemos encontrar en [RCC98] un nuevo sistema de coloración denominado $\ell\alpha\beta$. Este sistema trata de minimizar las correlaciones en escenas naturales comunes, basándose en el sistema visual humano. La baja correlación permite llevar a cabo transformaciones con la garantía de que usualmente no habrá errores.

El proceso a seguir es el siguiente: se transformará una imagen del sistema *RGB* al sistema $\ell\alpha\beta$, para más tarde estudiar como transferir en el sistema $\ell\alpha\beta$ la distribución de los colores de una imagen a otra.

Uno de los primeros sistemas de coloración desarrollados fue el **CIE xy**, y está compuesto por tres canales: X, la luminosidad; Y, la respuesta del ojo ante el color azul ; y Z, respuesta del ojo al color verde. Estos se parametrizan según:

$$x = \frac{X}{X+Y+Z}, \tag{3.1}$$

$$y = \frac{Y}{X+Y+Z}. \tag{3.2}$$

Utilizaremos este sistema porque no depende de ningún dispositivo. El blanco corresponde a los valores $x = y = \frac{1}{3}$, esto es, $X = Y = Z = 1$. Por lo tanto, trataremos de hacer corresponder al blanco del sistema CIE xy, el blanco del sistema RGB. Los canales de este último tienen los valores R=G=B=1. Tal y como vemos en [RAGS01], la matriz de conversión de RGB a XYZ es de la forma:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.5141 & 0.3239 & 0.1604 \\ 0.2651 & 0.6702 & 0.0641 \\ 0.0241 & 0.1228 & 0.8444 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \tag{3.3}$$

Una vez hemos convertido una imagen del sistema RGB al sistema CIE xy, convertimos este último al sistema de coloración natural del ojo: el sistema LMS. Este mide la respuesta de los tres tipos de conos, las células receptoras del color. Tiene, de nuevo, tres canales. La matriz de cambio del sistema CIE xy al sistema LMS es de la forma:

$$\begin{pmatrix} L \\ M \\ S \end{pmatrix} = \begin{pmatrix} 0.3897 & 0.6890 & -0.0787 \\ -0.2298 & 1.1834 & 0.0464 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}. \tag{3.4}$$

Al transformar los canales LMS a escala logarítmica, se reduce mucho la correlación entre ellos:

$$\mathbf{L} = \ln L, \qquad \mathbf{M} = \ln M, \qquad \mathbf{S} = \ln S. \tag{3.5}$$

Notamos esta transformación **LMS**, tal y como se indica. A partir de esto, [RCC98] optimiza la distribución del sistema **LMS** para incorrelar los canales, obteniendo así el sistema $\ell\alpha\beta$ tal y como sigue:

$$\begin{pmatrix} \ell \\ \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{L} \\ \mathbf{M} \\ \mathbf{S} \end{pmatrix}. \tag{3.6}$$

Para el proceso contrario, basta con invertir todas las matrices descritas en las anteriores

(a) RGB (b) XYZ (c) $\ell\alpha\beta$

Figura 3.2.: Imagen del logo de la Universidad de Granada en los sistemas de coloración RGB, XYZ y $\ell\alpha\beta$

ecuaciones, tomando potencias decimales en el paso de **LMS** a LMS. Podemos ver un ejemplo de los tres sistemas de color, RGB, XYZ y $\ell\alpha\beta$ en la Figura 3.2.

3.2. Transferencia de la distribución del color

Ahora trataremos de transferir los colores predominantes de una imagen a otra distinta. Una vez tenemos el resultado de minimizar las covarianzas, es suficiente con trabajar con las medias y las desviaciones típicas de los valores almacenados en los canales de cada píxel. Previamente, extraeremos las medias y las varianzas de cada uno de los canales $\ell\alpha\beta$, tanto de la imagen original como de la imagen cuyos colores transferiremos.

En primer lugar, centraremos los valores de cada canal:

$$\ell_1 = \ell - \bar{\ell} \quad (3.7) \qquad \alpha_1 = \alpha - \bar{\alpha} \quad (3.8) \qquad \beta_1 = \beta - \bar{\beta} \quad (3.9)$$

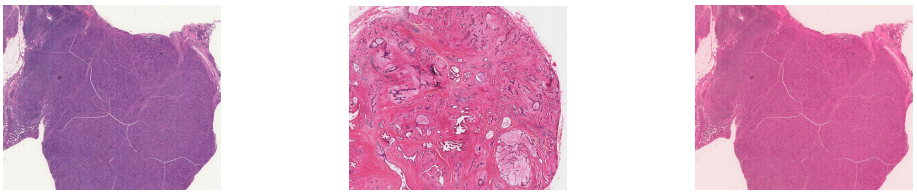
Ahora, basta con reescalar los canales centrados, estandarizando la original y escalándola según las desviaciones típicas de la imagen objetivo:

$$\ell_2 = \ell_1 \frac{\sigma^{\ell}_{\text{Original}}}{\sigma^{\ell}_{\text{Objetivo}}} \quad (3.10) \qquad \alpha_2 = \alpha_1 \frac{\sigma^{\alpha}_{\text{Original}}}{\sigma^{\alpha}_{\text{Objetivo}}} \quad (3.11) \qquad \beta_2 = \beta_1 \frac{\sigma^{\beta}_{\text{Original}}}{\sigma^{\beta}_{\text{Objetivo}}} \quad (3.12)$$

Así, por métodos sencillos, hemos conseguido que la imagen original tenga la misma desviación típica que la imagen objetivo. Para concluir, en lugar de sumarle la media de la imagen original, le sumamos la media de la imagen objetivo, obteniendo finalmente la misma distribución de colores en la imagen original.

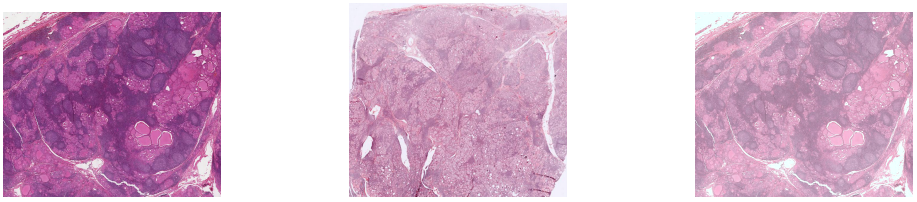
$$\ell_1 = \ell - \bar{\ell}_{\text{Objetivo}} \quad (3.13) \qquad \alpha_1 = \alpha - \bar{\alpha}_{\text{Objetivo}} \quad (3.14) \qquad \beta_1 = \beta - \bar{\beta}_{\text{Objetivo}} \quad (3.15)$$

Finalmente, utilizamos las transformaciones inversas a las descritas en Ecuación 3.3, Ecuación 3.4 y Ecuación 3.6, recordando el cambio logarítmico-potencial al pasar de LMS a **LMS**. La aplicación del proceso a imágenes reales se muestra en las Figuras 3.3, 3.4, 3.5 y 3.6.



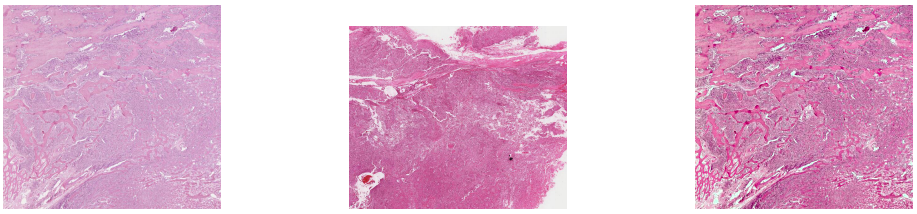
(a) Original (b) Color objetivo (c) Resultado

Figura 3.3.: Imagen histológica de cáncer de mama.



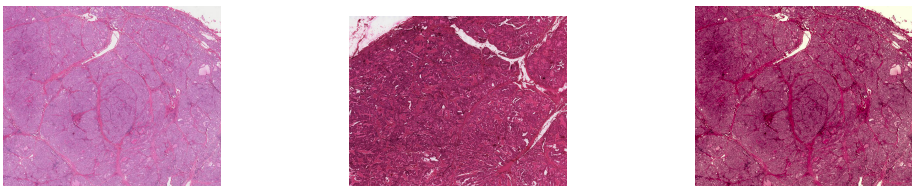
(a) Original (b) Color objetivo (c) Resultado

Figura 3.4.: Imagen histológica de tiroiditis de Hashimoto.



(a) Original (b) Color objetivo (c) Resultado

Figura 3.5.: Imagen histológica del osteosarcoma.



(a) Original (b) Color objetivo (c) Resultado

Figura 3.6.: Imagen histológica de la enfermedad de Graves.

4. Procesos gaussianos escalables variacionales para *crowdsourcing*.

En ámbitos como la Medicina, la cantidad de datos necesaria para problemas de clasificación se ve comprometida por el grado de conocimiento necesario para las etiquetas de las clases. Para ello se propone el *crowdsourcing*. El *crowdsourcing* es un nuevo método de obtención de etiquetas, en el que participantes con distintos grados de conocimiento sobre los datos aportan las etiquetas que los clasifican. Mediante él, se adquiere abundancia de información a cambio de perder certeza acerca de dichas anotaciones o etiquetas sobre las clases a las que pertenecen los datos. Este problema se ha afrontado desde el *deep learning*. Sin embargo, para tratar con grandes conjuntos de datos, se considera ventajosa la robusta estructura matemática de la clasificación mediante procesos gaussianos. Encontramos, sin embargo, un obstáculo en su desarrollo para el *crowdsourcing*. Los procesos gaussianos no pueden aplicarse sin realizar aproximaciones a grandes cantidades de datos. Es por eso que se propone una alternativa, los procesos gaussianos escalables, para tratar de dividir los datos en pequeños lotes, y mediante Inferencia Variacional, conseguir aproximaciones para el problema de clasificación.

4.1. Introducción.

Mientras que tradicionalmente se ha trabajado con datos anotados por profesionales, la posibilidad de sacrificar la correcta anotación a cambio de un mayor volumen de datos anotados resulta muy atractiva. Es de esa idea de la que parte el *crowdsourcing*. Este método de trabajo está siendo ampliamente aplicado en áreas como las imágenes médicas o estudios genéticos, entre otros. En él, se cuenta con participantes con diversos grados de conocimiento sobre los datos, para que establezcan las etiquetas de los datos. Por tanto se cuenta con distintas etiquetas para un mismo conjunto de rasgos, de distinto grado de verosimilitud, sobre las que hay que hacer inferencia. Por eso, puede ser considerado como un problema de clasificación no supervisado.

Este método plantea una nueva problemática: hay que tratar de combinar la falta de certeza sobre la corrección de las anotaciones con la falta de acuerdo entre los anotadores, así como tratar de detectar anotadores que den anotaciones falsas o erróneas. Entre estos últimos destacan los **anotadores *spam*** y los **anotadores adversarios**. Los anotadores *spam* proporcionan etiquetas independientes de los datos. Podrían dar siempre la misma etiqueta, o elegir una al azar, por ejemplo. Por otro lado, los anotadores adversarios siempre proporcionan una respuesta errónea. Estos últimos han de tener información sobre cuál era la verdadera, puesto que han de dar otra distinta. Por lo tanto, estos sí que proporcionan etiquetas que dependen de los datos observados.

En una primera aproximación, parece sensato suponer a priori una misma corrección en todos los anotadores. Otra aproximación más compleja puede tratar de calibrar el sesgo

de cada anotador. En ese caso, se puede tratar de encontrar los anotadores más correctos, y proceder a un problema de clasificación estándar, de no *crowdsourcing*, suponiendo una etiqueta verdadera para cada dato.

Estudios recientes prueban que los resultados son mejores cuando se trata de modelar al mismo tiempo los anotadores y los datos. Para más información, puede acudirse al artículo sobre el que versa este capítulo, [MARC+20]. Es aquí donde surge la necesidad de encontrar un método matemáticamente robusto como los procesos gaussianos (GP), que además sea capaz de devolver resultados muy ajustados sobre la incertidumbre de los anotadores. El principal problema que presentan los GPs es que no pueden trabajar con grandes volúmenes de datos. Es por ello que trataremos de hacerlos escalables mediante unos puntos, llamados inductores, que resumen la información de todo el conjunto de datos. El objetivo será construir un método que pueda adaptarse a los datos en pequeños lotes (*mini-batches*) de manera que pueda llegar a ser usado con cualquier volumen de datos, sea lo grande que sea.

Veremos que el ELBO obtenido por el modelo descrito es de un tipo muy parecido al desarrollado en la Subsección 1.4.2, y que por tanto permite un tratamiento numérico de la aproximación de la distribución a posteriori.

Para tratar con la incertidumbre de los anotadores, utilizaremos inferencia bayesiana utilizando la extensión mutivariante de la distribución estudiada en el Subsección 1.2.1: trataremos con una distribución multinomial en lugar de la binomial, y distribución Dirichlet a priori en lugar de la distribución beta.

Finalmente, se cerrará el capítulo y el trabajo con un ejemplo sobre lo desarrollado en este capítulo, que aúna todo lo desarrollado en los capítulos anteriores.

4.2. Punto de partida del modelo.

Trataremos con un problema de *crowdsourcing* de K clases. Los datos observados se describen en el conjunto $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{Y}_n^a) | a \in A_n; n = 1, \dots, N\}$, donde:

- $\mathbf{x}_n \in \mathbb{R}^D$ es el vector de características observadas en cada dato.
- $\mathbf{Y}_n^a$ son las anotaciones realizadas para el n-ésimo dato según el anotador a. Por tanto, $\mathbf{Y}_n^a$ es un subconjunto del conjunto de vectores $\{\mathbf{e}_1, \dots, \mathbf{e}_K\}$, donde el vector $\mathbf{e}_i$ vale 0 en todas las componentes, menos en la i-ésima, que vale 1. Así, si el anotador a determina que el dato n pertenece a la clase k, $\mathbf{Y}_n^a = \{\mathbf{e}_k\}$.
- Se trata de una muestra de tamaño N, con A anotadores. A_n denota el conjunto de los anotadores que han etiquetado el n-ésimo dato.
- Notaremos como $\mathbf{X}$ al conjunto de todos los datos observados.
- Notaremos como $\mathbf{Y}$ al conjunto de todas las anotaciones.
- Notaremos $\mathbf{z}_n$ a las anotaciones reales o teóricas de cada dato, que son desconocidas.
- Las etiquetas asignadas por los anotadores $\mathbf{Y}$ dependen de las anotaciones reales $\mathbf{z}_n$ y el grado de credibilidad de cada anotador, modelado según la **matriz de confusión** $\mathbf{R}^a \in \mathcal{M}_K$. Se trata de una matriz estocástica, esto es, cada columna suma 1, con elementos en $[0, 1]$. Cada elemento de la matriz $(\mathbf{R}^a)_{ij}$ representa la probabilidad de

que el anotador a etiquete en la clase i-ésima un dato de la clase j-ésima. Por tanto:

$$p(\mathbf{Y}_n^a|\mathbf{z}_n, \mathbf{R}^a) = \prod_{\mathbf{y}\in\mathbf{Y}_n^a} \mathbf{y}^T \mathbf{R}^a \mathbf{z}_n \tag{4.1}$$

- Se considera que los anotadores etiquetan cada dato de manera independiente. Tomando $\mathbf{Z}$ como el conjunto de las anotaciones reales y $\mathbf{R}$ como el conjunto de las matrices de confusión, obtenemos:

$$p(\mathbf{Y}|A, \mathbf{R}) = \prod_{n=1}^{N} \prod_{a\in A_n} p(\mathbf{Y}_n^a|\mathbf{z}_n, \mathbf{R}^a). \tag{4.2}$$

- Dada la naturaleza del problema de determinar las etiquetas correctas, para cada dato tenemos un problema de inferencia de una distribución categórica. Así, se modelará según lo desarrollado en el Capítulo 1, tomando una distribución a priori Dirichlet por ser la distribución a priori conjugada de la distribución categórica. Si notamos como $(\mathbf{R}^a)_j$ a la columna j-ésima de la matriz de confusión del anotador a, tenemos una distribución a priori sobre $\mathbf{R}$:

$$p(\mathbf{R}) = \prod_{a=1}^{A} p(\mathbf{R}^a) = \prod_{a=1}^{A}\prod_{j=1}^{K} p((\mathbf{R}^a)_j), \qquad (\mathbf{R}^a)_j \rightsquigarrow \mathrm{Dir}(\alpha_{1j}^a, \ldots, \alpha_{Kj}^a) \tag{4.3}$$

  Como determinábamos en dicho capítulo, en caso de no disponer de información a priori, basta con tomar una distribución uniforme, esto es, $\alpha_{1j}^a = \cdots = \alpha_{Kj}^a = 1, \quad \forall j = 1, \ldots, K$.

- Cada distribución teórica de los datos $\mathbf{f}_k, \forall k = 1, \ldots, K$ se modela según un GP a priori con *kernel* una RBF con hiperparámetros $\boldsymbol{\theta}_k = (\sigma_k, \ell_k)$. Así, tenemos K variables latentes modelando cada una de las clases. Notando Θ al conjunto de todos los hiperparámetros, obtenemos:

$$\mathbf{f}_k|\Theta = \boldsymbol{\theta}_k, \mathbf{X} \rightsquigarrow \mathcal{N}(0, K(\boldsymbol{\theta}_k; \mathbf{X})), \tag{4.4}$$

$$p(\mathbf{F}|\Theta, \mathbf{X}) = \prod_{k=1}^{K} p(\mathbf{f}_k|\Theta = \boldsymbol{\theta}_k, \mathbf{X}). \tag{4.5}$$

- Al tratarse de un problema de clasificación por modelos lineales, trataremos de modelar la verosimilitud, como se exponía en la Sección 2.4. Así, se busca un vector $\nu(\mathbf{f}_1, \ldots, \mathbf{f}_K) \in [0,1]^K$ con $\|\nu(\mathbf{f}_1, \ldots, \mathbf{f}_K)\|_1 = 1$. Lo que se busca por tanto es:

$$p(z|\mathbf{f}_1, \ldots, \mathbf{f}_K) = z^T \nu(\mathbf{f}_1, \ldots, \mathbf{f}_K). \tag{4.6}$$

  Dicha función $\nu(\mathbf{f}_1, \ldots, \mathbf{f}_K)$ se elige ajustándose a cada problema. Veremos un ejemplo de ello en la Sección 4.5.

- De nuevo, notamos $\mathbf{F}$ al conjunto de las distribuciones teóricas de todos los datos. Se trata por tanto de una matriz $\mathbf{F} \in \mathcal{M}_{N\times K}$ donde cada término $\mathbf{F}_{ij}$ es el valor de la k-ésima variable en el dato n.

- Asumimos que las etiquetas reales de cada dato son independientes, de manera que se obtiene:

$$p(\mathbf{Z}|\mathbf{F}) = \prod_{n=1}^{N} p(\mathbf{z}_n|\mathbf{f}_{n,:}). \tag{4.7}$$

  Se ha utilizado la notación $\mathbf{f}_{n,:}$ para denotar la fila n-ésima, de manera diferenciada de $\mathbf{f}_k$ para la columna k-ésima.

Finalmente, en resumen, dada la Ecuación 4.2,Ecuación 4.3, Ecuación 4.7 y Ecuación 4.5, el modelo probabilístico que resolveremos es el siguiente:

$$p(\mathbf{Y}, \mathbf{Z}, \mathbf{F}, \mathbf{R}|\Theta) = p(\mathbf{Y}|\mathbf{Z}, \mathbf{R})p(\mathbf{R})p(\mathbf{Z}|\mathbf{F})p(\mathbf{F}|\Theta), \tag{4.8}$$

donde se ha omitido la dependencia a los datos $\mathbf{X}$ por claridad.

4.3. Procesos gaussianos escalables

Una vez sentadas las bases del modelo, trataremos ahora de hacer un inciso para llegar a conseguir un proceso gaussiano escalable (SGP). Para ello tomaremos M puntos inductores para cada proceso gaussiano a priori $\mathbf{f}_k$, que resuman la información de la distribución. Así, podemos considerar un nuevo proceso gaussiano que considere toda la información previa y de manera marginal la distribución predictiva sobre estos puntos. Esta se desarrollaba en la Ecuación 2.35. Notaremos a los M puntos inductores, $\tilde{\mathbf{X}}_k$, cuya distribución es un GP, $\mathbf{u}_k$, de manera análoga a como $\mathbf{f}_k$ es la distribución de $\mathbf{x}_k$. Como se trata de un proceso gaussiano con un conjunto finito de índices, no es más que una distribución normal M-multivariante. Podemos entonces considerar la distribución conjunta $(\mathbf{f}_k, \mathbf{u}_k)$, factorizando como sigue:

$$p(\mathbf{f}_k, \mathbf{u}_k) = p(\mathbf{f}_k|\mathbf{u}_k)p(\mathbf{u}_k). \tag{4.9}$$

Como veníamos haciendo en la sección anterior, notamos $\mathbf{U} \in \mathcal{M}_{M\times K}$ a la matriz que tiene por columnas las distribuciones de los puntos inductores, tal y como notábamos $\mathbf{F} \in \mathcal{M}_{N\times K}$ a todas los GPs a priori. Podemos reescribir ahora la Ecuación 4.8 como sigue:

$$p(\mathbf{Y}, \mathbf{Z}, \mathbf{F}, \mathbf{U}, \mathbf{R}|\Theta) = p(\mathbf{Y}|\mathbf{Z}, \mathbf{R})p(\mathbf{R})p(\mathbf{Z}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \Theta)P(\mathbf{U}|\Theta). \tag{4.10}$$

La escalabilidad de este proceso gaussiano, esto es, la posibilidad de llevar a cabo la clasificación utilizando pequeños lotes de datos en lugar de todos los datos a la vez, se pondrá de manifiesto en la siguiente sección. Una vez corregida dicha escalabilidad, se presentan dos alternativas de solución al problema: podemos tratar de encontrar un problema aproximadamente igual y llevar a cabo inferencia clásica o bayesiana; o bien, podemos mantener el modelo inalterado, tratando de encontrar una solución aproximada. En este capítulo desarrollamos la segunda opción, en la que se utiliza inferencia variacional para tratar de inferir la distribución teórica de manera aproximada, tal y como se expone en el Capítulo 1.

4.4. Procesos gaussianos escalables variacionales.

Desarrollaremos ahora las aproximaciones, mediante inferencia variacional, que son necesarias para la adaptación de los procesos gaussianos escalables al *crowdsourcing*. A esto lo llamaremos procesos gaussianos escalables variacionales para *crowdsourcing* (SVGPCR). La idea de recurrir a aproximaciones surge de que el cálculo de la verosimilitud marginal, $p(\mathbf{Y}|\Theta)$ es intratable, pero es necesaria para la optimización de los hiperparámetros del *kernel* Θ, y así poder aproximar la distribución a posteriori, tal y como exponíamos en el Capítulo 1.

Ahora, tal y como vemos en [MARC+20], la distribución a posteriori aproximada la buscaremos siguiendo los siguientes criterios, tal y como hacíamos al considerar la familia *mean field* en Subsección 1.4.2:

$$q(\mathbf{Z},\mathbf{F},\mathbf{U},\mathbf{R}) = q(\mathbf{Z})q(\mathbf{F}|\mathbf{U},\Theta)q(\mathbf{U})q(\mathbf{R}), \text{donde :} \tag{4.11}$$

$$q(\mathbf{Z}) = \prod_{n=1}^{N} q(\mathbf{z}_n) = \prod_{n=1}^{N} \mathbf{z}_n^T \mathbf{q}_n, \tag{4.12}$$

$$q(\mathbf{F}|\mathbf{U},\Theta) = p(\mathbf{F}|\mathbf{U},\Theta), \tag{4.13}$$

$$q(\mathbf{U}) = \prod_{k=1}^{K} q(\mathbf{u}_k), \quad \text{donde} \quad \mathbf{u}_k \rightsquigarrow \mathcal{N}(\mathbf{m}_k, \mathbf{S}_k), \tag{4.14}$$

$$q(\mathbf{R}) = \prod_{k=1}^{K}\prod_{a=1}^{A} q(\mathbf{r}_k^a). \tag{4.15}$$

Hagamos algunas apreciaciones:

- En la Ecuación 4.12, cada $\mathbf{q}_n = (q_{1n},\dots,q_{Kn}) \in [0,1]^K$ es la probabilidad de que el dato n pertenezca a cada una de las K clases. Es una aproximación *mean field*.
- En la Ecuación 4.13, simplemente consideramos la distribución a priori condicionada. En esto es esencial la independencia entre los puntos inductores y los observados.
- En la Ecuación 4.14, cada $\mathbf{m}_k \in \mathbb{R}^M$ es la media de la distribución y $\mathbf{S}_k \in \mathcal{M}_M$ es la matriz definida positiva de covarianzas. Es una aproximación *mean field*.
- En la Ecuación 4.14, para cada $\mathbf{r}_k^a$ buscaremos una distribución de Dirichlet de parámetros $\mathrm{Dir}(\tilde{\alpha}_{k1}^a,\dots,\tilde{\alpha}_{kK}^a)$, de manera que se ajuste a la distribución a priori tal y como se describía en la Ecuación 4.3. Es una aproximación *mean field*.
- De ahora en adelante, notaremos como V al conjunto de parámetros descritos anteriormente: $\{\mathbf{q}_i : i = 1,\dots,N\}, \{\mathbf{m}_i, \mathbf{S}_i : i = 1,\dots,K\}, \{\tilde{\alpha}_{ij}^a : i,j = 1,\dots,K, a = 1,\dots,A\}$.

Podemos utilizar la Proposición 1.6 para calcular la verosimilitud marginal:

$$\ln p(\mathbf{Y}|\Theta) = \mathrm{KL}(q(\mathbf{Z},\mathbf{F},\mathbf{U},\mathbf{R})||p(\mathbf{Z},\mathbf{F},\mathbf{U},\mathbf{R}|\mathbf{Y},\Theta)) + \mathrm{ELBO}(q). \tag{4.16}$$

Ahora es posible calcular de manera explícita la expresión del ELBO, a partir de la Def. 1.12,

recordando la Ecuación 4.10 y la Ecuación 4.11:

$$\text{ELBO}(q) = \mathbb{E}_{q(\mathbf{Z})q(\mathbf{F}|\mathbf{U})q(\mathbf{U})q(\mathbf{R})}\left[\ln \frac{p(\mathbf{Y}|\mathbf{Z},\mathbf{R})p(\mathbf{Z}|\mathbf{F})\cancel{p(\mathbf{F}|\mathbf{U})}p(\mathbf{U})p(\mathbf{R})}{q(\mathbf{Z})\cancel{q(\mathbf{F}|\mathbf{U})}q(\mathbf{U})q(\mathbf{R})}\right] \tag{4.17}$$

$$= \mathbb{E}_{q(\mathbf{Z})q(\mathbf{R})}[\ln p(\mathbf{Y}|\mathbf{Z},\mathbf{R})] + \mathbb{E}_{q(\mathbf{Z})q(\mathbf{F}|\mathbf{U})q(\mathbf{U})}[\ln p(\mathbf{Z}|\mathbf{F})] - \mathbb{E}_{q(\mathbf{Z})}[\ln q(\mathbf{Z})]$$
$$+ \mathbb{E}_{q(\mathbf{U})}\left[\ln \frac{p(\mathbf{U})}{q(\mathbf{U})}\right] + \mathbb{E}_{q(\mathbf{R})}\left[\ln \frac{p(\mathbf{R})}{q(\mathbf{R})}\right] \tag{4.18}$$

$$= \mathbb{E}_{q(\mathbf{Z})q(\mathbf{R})}[\ln p(\mathbf{Y}|\mathbf{Z},\mathbf{R})] + \mathbb{E}_{q(\mathbf{Z})q(\mathbf{F}|\mathbf{U})q(\mathbf{U})}[\ln p(\mathbf{Z}|\mathbf{F})] - \mathbb{E}_{q(\mathbf{Z})}[\ln q(\mathbf{Z})]$$
$$- \sum_{k=1}^{K} \text{KL}(q(\mathbf{u}_k)||p(\mathbf{u}_k)) - \sum_{a=1}^{A}\sum_{k=1}^{K} \text{KL}(q(\mathbf{r}_k^a)||p(\mathbf{r}_k^a)). \tag{4.19}$$

Los resultados de lo desarrollado a lo largo del trabajo pueden verse en esta operación. Algunos detalles que lo ilustran:

- Vemos en el paso Ecuación 4.17 las consecuencias de lo desarrollado en la Sección 4.3: sin ellas, no podríamos cancelar los términos. Esto es lo que determina la escalabilidad del método, puesto que podemos expresar todos los términos del ELBO en función de V, Θ y $\tilde{\mathbf{X}}$.
- El único término que no depende exclusivamente de dichos parámetros es

  $$\mathbb{E}_{q(\mathbf{Z})q(\mathbf{F}|\mathbf{U})q(\mathbf{U})}[\ln p(\mathbf{Z}|\mathbf{F})],$$

  en la Ecuación 4.17. Sin embargo, podemos conseguir una dependencia exclusiva llevando a cabo aproximaciones sencillas.
- Además, este término es la esperanza del logaritmo de la verosimilitud de una distribución normal. Como veíamos en el Capítulo 2, en los casos de regresión lineal este es analíticamente tratable. Sin embargo, estamos tratando con un método de clasificación. Como veíamos en la Sección 2.4, la verosimilitud de un problema de clasificación por métodos lineales se aborda por métodos numéricos, ya que en general, es analíticamente intratable.
- El término $\mathbb{E}_{q(\mathbf{Z})q(\mathbf{R})}[\ln p(\mathbf{Y}|\mathbf{Z},\mathbf{R})]$ en la Ecuación 4.17 no es más que la esperanza del logaritmo de la distribución Dirichlet. En este paso, resulta esencial que la distribución Dirichlet sea la distribución a priori conjugada de la distribución categórica, tal y como se demostraba para el caso univariable en la Subsección 1.2.1.
- En la Ecuación 4.19, encontramos dos KL divergencias.
  - La primera, $\text{KL}(q(\mathbf{u}_k)||p(\mathbf{u}_k))$, es la KL divergencia entre dos normales. En este caso son mutivariantes, luego es la extensión a mutivariante del ejemplo desarrollado en la Subsubsección 1.4.1.1. En concreto, si notamos $K = K(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})$ a la matriz de covarianzas entre los puntos inducidos, lo que obtenemos es:

    $$\text{KL}(\mathcal{N}(\mathbf{m}_k, \mathbf{S}_k)||\mathcal{N}(0, \mathbf{K})) = \frac{1}{2}\left(\ln\left(\frac{\det \mathbf{K}}{\det \mathbf{S}_k}\right) + \text{tr}(\mathbf{K}^{-1}\mathbf{S}_k) + \mathbf{m}_k^T\mathbf{K}^{-1}\mathbf{m}_k - M\right)$$

  - La segunda, $\text{KL}(q(\mathbf{r}_k^a)||p(\mathbf{r}_k^a))$ es la KL divergencia entre dos distribuciones Dirichlet. Podemos encontrar más detalles sobre su desarrollo en [MARC+20].

Finalmente, vemos que la expresión del ELBO está factorizada según los datos observados. Esto nos permite trabajar con lotes de datos de menor volumen o *mini-batches*, e ir ajustando el modelo de manera iterativa, por lotes. Esto se utilizará para llevar a cabo una aproximación estocástica de la ELBO. Por tanto, hemos solucionado el problema del método de *crowdsourcing* mediante un método que es capaz de enfrentarse a conjuntos de datos tan grandes como sean necesarios.

Llegados a este punto, y siguiendo la estructura que se ha desarrollado a lo largo del trabajo, concluimos con el cálculo de la densidad predictiva del modelo.

4.4.1. Densidad predictiva de los procesos gaussianos escalables variacionales en *crowdsourcing*.

Una vez se ha optimizado el ELBO por métodos numéricos, y quedan determinados V, Θ y $\tilde{\mathbf{X}}$, podemos llevar a cabo el cálculo de nuevas predicciones utilizando la Ecuación 1.9:

$$p(\mathbf{f}_k^*|\mathbf{x}^*, \mathcal{D}) = \int p(\mathbf{f}_k^*|\mathbf{u}_k)p(\mathbf{u}_k|\mathcal{D})\mathrm{d}\mathbf{u}_k = \int p(\mathbf{f}_k^*|\mathbf{u}_k)q(\mathbf{u}_k)\mathrm{d}\mathbf{u}_k. \tag{4.20}$$

Tal y como vemos en [Bro13], esto quiere decir que:

$$\begin{aligned}\mathbf{f}_k^*|\mathbf{x}^*, \mathcal{D} \rightsquigarrow \mathcal{N}(&K(\mathbf{x}^*, \tilde{\mathbf{X}})K(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})^{-1}\mathbf{m}_k,\\ &K(\mathbf{x}^*, \mathbf{x}^*) + K(\mathbf{x}^*, \mathbf{X}^*)K(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})^{-1}(\mathbf{S}_k - K(\tilde{\mathbf{X}}, \tilde{\mathbf{X}})K(\tilde{\mathbf{X}}, \mathbf{x}^*)K(\mathbf{x}^*, \mathbf{x}^*)^{-1}).\end{aligned} \tag{4.21}$$

Además, podemos calcular la densidad predictiva de la etiqueta real $\mathbf{z}^*$ de dicha predicción. Utilizamos de nuevo la Ecuación 1.9, una vez tenemos una predicción:

$$p(\mathbf{z}^*) = \int p(\mathbf{z}^*|\mathbf{f}^*)p(\mathbf{f}^*)\mathrm{d}\mathbf{f}^*. \tag{4.22}$$

De nuevo, esta integral es normalmente calculada por métodos numéricos. Además, las distribuciones a posteriori aproximadas, $q(\mathbf{Z})$ y $q(\mathbf{R})$ nos dan información acerca de cómo de correctas eran las etiquetas aportadas por los anotadores.

4.5. Aplicación de los SVGPCR a imágenes histológicas de cáncer de mama

Tal y como se mencionaba en la sección anterior, mediante *crowdsourcing* se trata de llevar a cabo clasificación utilizando etiquetas con ruido, es decir, anotadores cuyo grado de conocimiento sobre las observaciones se desconoce. Esto tiene especial importancia en campos como la Medicina, donde contar con etiquetas de expertos patólogos es difícil, y hace que estos no dispongan de ese tiempo para otro tipo de tareas médicas. Para terminar el trabajo, analizaremos el artículo [LPAMÁ+21] en el que han desarrollado distintos métodos de clasificación por SVGP, donde analizan si los SVGPCR pueden aliviar la tarea de anotación de los expertos patólogos, en tareas de investigación biomédica. Para ello, trabajan con el banco de imágenes histológicas de cáncer de mama [AEH+19].

La clasificación trata de establecer qué área de la imagen corresponde con estroma, el tejido conjuntivo, qué área de la imagen corresponde con tumor canceroso y qué área corres-

ponde con linfocitos, células que se infiltran en el tumor canceroso para eliminarlo. Para ello contarán con 2 expertos patólogos y 20 estudiantes de medicina que anotarán los datos. Se tomarán 161 imágenes, en las que se seleccionará una región de interés o ROI por sus siglas en inglés, en las que se llevarán a cabo las anotaciones.

Las anotaciones de los expertos patólogos servirán para evaluar los resultados obtenidos mediante los tres métodos de clasificación utilizados. Con las anotaciones de los estudiantes de medicina, se establecen dos grupos: por un lado, se utilizan 10 imágenes que son anotadas por todos los participantes, para llevar a cabo evaluaciones (ROIs de evaluación); y por otro lado, las 151 imágenes restantes se reparten entre todos (ROIs principales).

Para clasificarlas, se utilizan tres métodos, expuestos a continuación. En primer lugar, se toma la moda de las anotaciones. En muchos casos, esto suele disminuir el ruido de las anotaciones. Así, se lleva a cabo una clasificación por SVGP, a la que llamamos SVGP por mayoría. En segundo lugar, se utilizan todas las anotaciones de los estudiantes. Aunque tomar la moda de las anotaciones reduce el ruido, en Medicina no suele ser buena idea alterar las anotaciones obtenidas, por lo que es mejor determinar el grado de conocimiento de quien las ha realizado, y mantenerlas inalteradas. Así, se lleva a cabo una clasificación mediante SVGPCR. Finalmente, se cuenta con ayuda de los expertos para corregir las anotaciones de los estudiantes, y se utilizan las anotaciones de los estudiantes corregidas junto con las de los expertos patólogos. Al contar con que las anotaciones son realizadas con un alto grado de conocimiento sobre los datos, se lleva a cabo una clasificación supervisada mediante SVGP. La llamamos SVGP *gold*.

El color de las imágenes ha sido normalizado antes de llevar a cabo la clasificación, tal y como se introdujo en el Capítulo 3. Tras esto, mediante *deep learning*, se han extraído 516 rasgos o características que se estudian en cada imagen. Para mayor información sobre *deep learning*, se puede consultar [GBC16].

La clasificación por tanto lleva a cabo 3 anotaciones distintas: tumor, estroma y linfocitos (**Y**). Se estudian 516 características(**X**).

En el experimento de *crowdsourcing*, se cuenta con 20 anotadores (A). Además, para $\nu(f_1, \ldots, f_k)$ (Ecuación 4.6) se utiliza la verosimilitud soft-max, siguiendo con la notación desarrollada en la sección anterior:

$$p(z_n = e_k | f_{n,:}) = \frac{\exp(f_{n,k})}{\sum_{c=1}^{K} \exp(f_{n,c})}. \tag{4.23}$$

La función kernel utilizada es una RBF de hiperparámetros σ y ℓ. Se toman distintos vectores de las 512 características como puntos inductores para hacer el proceso escalable.

Para concluir, introduciremos conceptos de análisis de resultados para aprendizaje automático, basándonos en [Gé17]. Para ello, llamamos **conjunto de entrenamiento** al conjunto de anotaciones sobre las que se ajusta el modelo de clasificación. Llamamos **conjunto test** al conjunto de anotaciones de los expertos, que se utilizarán como las anotaciones verdaderas (**Z**) para llevar a cabo estas mediciones. Así, las mediciones se hacen para clasificación binaria, y para su extensión a problemas de clasificación multiclase basta con hacer la media, considerando un problema de clasificación binaria para cada clase: se toma como verdadero pertenecer a dicha clase, y como falso no pertenecer a dicha clase. El razonamiento sigue la filosofía de lo descrito en el E.j. 2.7. Comenzamos con unas definiciones básicas:

Definición 4.1. Sea y^* la anotación predicha por un modelo de clasificación binario, sea y la etiqueta real, y sean las etiquetas $Y = \{1, -1\}$. Se considera **resultado positivo** la etiqueta 1 y **resultado negativo** la etiqueta -1 :

- Decimos que un resultado es un **verdadero positivo** o VP si $y^* = 1, y = 1$
- Decimos que un resultado es un **falso positivo** o FP si $y^* = 1, y = -1$
- Decimos que un resultado es un **verdadero negativo** o VN si $y^* = -1, y = -1$
- Decimos que un resultado es un **falso negativo** o FN si $y^* = -1, y = 1$

Definición 4.2. Llamamos **tasa de acierto** o precisión a

$$p = \frac{VP}{VP + FP}, \tag{4.24}$$

esto es, a la tasa de verdaderos positivos.

Definición 4.3. Llamamos **sensibilidad** a

$$s = \frac{VP}{VP + FN}, \tag{4.25}$$

esto es, a la tasa de positivos detectados.

Definición 4.4. Llamamos **especificidad** a

$$sp = \frac{TN}{VN + FN}, \tag{4.26}$$

esto es, a la tasa de verdaderos negativos.

Definición 4.5. Llamamos **puntuación F1** a la media armónica de la tasa de acierto y la sensibilidad:

$$F1 = \frac{2}{\frac{1}{p} + \frac{1}{s}} = 2\frac{p \cdot s}{p + s} = \frac{VP}{VP + \frac{FN+FP}{2}}. \tag{4.27}$$

Como vemos, una alta puntuación F1 se obtiene si el modelo de clasificación tiene una tasa de acierto y una sensibilidad similares. Sin embargo, esto no es siempre deseable: en el caso del cáncer de mama, por razones médicas, es beneficioso tener una mayor sensibilidad que tasa de acierto, puesto que una mayor tasa de positivos detectados es más ventajosa que una mayor tasa de verdaderos positivos.

De hecho, la necesidad de aumentar una de las dos medidas pasa por asumir la disminución de la otra. Podríamos estudiar la curva que se obtiene al representar la tasa de acierto frente a la sensibilidad. Sin embargo, otra curva es más descriptiva y ampliamente utilizada:

Definición 4.6. Llamamos curva de la característica operante del receptor o **curva ROC** a la representación de la sensibilidad frente a la tasa de falsos negativos, esto es, s frente a $1 - sp$.

Esta curva describe cuantos falsos negativos estamos dispuestos a aceptar frente a cuantos positivos se detectan. Así, en el caso del cáncer de mama, se tratará de que el modelo de clasificación tenga una alta sensibilidad, esto es por tanto, minimizar los falsos negativos. Una forma de medir esto es calcular el área bajo la curva ROC, que notamos como AUC por sus siglas en inglés. Un clasificador perfecto tendrá un AUC$= 1$, mientras que uno totalmente aleatorio tendrá un AUC$= 0.5$.

Hechas estas definiciones, estamos en disposición de analizar los resultados expuestos en el artículo [LPAMÁ+21], tal y como vemos en la Tabla 4.1.

	Puntuación F1	Tasa de acierto	AUC
SVGP *gold*	0.8157	0.8582	0.9373
SVGP por mayoría	0.7919	0.8458	0.9289
SVGPCR	0.8147	0.8579	0.9360

Tabla 4.1.: Medidas de acierto de los resultados obtenidas en [LPAMÁ+21] para las tres clasificaciones por procesos gaussianos escalables variacionales llevadas a cabo.

Vemos que el SVGP *gold* tiene una mayor puntuación F1, precisión y AUC comparado con el SVGP por mayoría. En concreto, SVGP obtiene una puntuación F1 un 3 % mayor. Además, se obtiene un AUC un 0.9 % mayor y una precisión un 1.5 % mayor. Sin embargo, los resultados obtenidos entre SVGPCR y SVGP *gold* son realmente similares, con puntuaciones F1 casi idénticas (0.815 frente a 0.816), al igual que las AUC (0.936 frente a 0.937) y prácticamente la misma precisión. Así, las diferencias entre la clasificación utilizando *crowdsourcing* son mucho menores que las obtenidas mediante voto mayoritario.

Tal y como se menciona en el artículo [LPAMÁ+21], la matriz de confusión también aporta información: por ejemplo, el estudiante número 17 tendía a anotar el estroma como tumor, mientras que el estudiante número 19 era menos sensible a la detección del tumor.

Tomando predicciones para cada modelo y superponiendo lo obtenido con los datos del conjunto test, se puede tomar el área coincidente como medida de la bondad del modelo.

Definición 4.7. Dado un modelo de clasificación de imágenes, al superponer una imagen con la predicción llevada a cabo por el modelo sobre los datos de la misma imagen, obtenemos áreas en los que las anotaciones coinciden a las que notamos C, y en las que las anotaciones no coinciden, a las que llamamos NC. Se llama **coeficiente DICE** a lo siguiente:

$$DICE = \frac{2C}{NC + 2C}. \tag{4.28}$$

Proposición 4.1. *El coeficiente DICE equivale a la puntuación F1.*

Demostración. Basta observar que C=VP y que NC=FP+FN, de manera que:

$$DICE = \frac{2C}{NC + 2C} = \frac{2VP}{FP + FN + 2VP} = \frac{VP}{VP + \frac{FN+FP}{2}} = F1. \tag{4.29}$$

□

Así, tal y como encontramos en el artículo [LPAMÁ+21], podemos establecer un intervalo de confianza para la puntuación F1, 0.7789 ± 0.0237, de manera que como vemos en la Tabla 4.1, la puntuación F1 obtenida está fuera de dicho intervalo.

Con este resultado concluimos el trabajo, observando la potencia de los resultados obtenidos mediante los SVGPCR frente al problema de clasificación supervisada SVGP *gold*. En este caso, vemos que se puede carecer de las correcciones llevadas a cabo por los expertos sin sacrificar la bondad del modelo. Los modelos se han desarrollado a partir de la información de que los anotadores sí que tenían un grado de conocimiento sobre los datos avanzado. Además, las áreas correspondientes a tumores y estromas tenían cotas claras, lo que ha ayudado en la tarea de anotación, dado que los anotadores han tendido a coincidir. Además, utilizando los SVGPCR hemos conseguido medir el sesgo con el que los anotadores llevaban a cabo las anotaciones, lo que podría permitir utilizar los SVGPCR para asignar a cada anotador un

trabajo que se ajuste a sus tendencias, o utilizarlo para mejorar la precisión de sus anotaciones. Además, ha sido necesario utilizar *deep learning* para la correcta acotación de las áreas de interés, por lo que este modelo podría ser adecuado para estudiar imágenes que presenten parches y cuya clasificación dependa de ellos. Tal y como concluye el artículo [LPAMÁ+21] (al que se refiere para mayor información y conclusiones más profundas), quedan problemas abiertos, como entender cómo llevar a cabo mediante la matriz de confusión las mejoras en los anotadores o cómo afectan los rasgos utilizados en los resultados obtenidos.

4.6. Conclusión y futuro trabajo.

En primer lugar, se han desarrollado los conceptos básicos de inferencia bayesiana. Dado que la inferencia exacta da problemas para su cálculo analítico en situaciones reales, se ha desarrollado la teoría de inferencia variacional para tratar de aproximar distribuciones mediante un problema de optimización de Análisis Funcional. Así, la inferencia variacional se muestra como una herramienta potente para llevar a cabo inferencia aproximada mediante métodos numéricos, pese a la dificultad que entrañe cada problema.

Hemos utilizado la inferencia variacional para hacer inferencia en el método de clasificación por procesos gaussianos para *crowdsourcing*. El experimento analizado demuestra que la teoría desarrollada, además de ser matemáticamente robusta, da buenos resultados. No se han analizado en este trabajo otros métodos de inferencia más ampliamente utilizados. Por ejemplo, el método de cadenas de Markov Monte Carlo o MCMC, el cual puede ser más preciso pero también supone un mayor coste computacional. Tampoco se han estudiado métodos de clasificación basados en jerarquías de procesos gaussianos, también llamados procesos gaussianos profundos. Estos podrían obtener resultados más precisos ya que el clasificador sería más flexible.

Además, tal y como se menciona en la última sección, el tratamiento de la información de la matriz de confusión es esencial. Podría plantearse la incorporación de la dificultad que entraña cada muestra en la propia matriz de confusión. Así, podría corregirse la falta de conocimiento del anotador, debido a que la dificultad de las muestras en una misma clase puede ser diferente. Esto también podría mejorar los resultados del método aquí propuesto.

El método de normalización de color desarrollado tan solo da una idea del tratamiento de imágenes previo a una tarea de clasificación. Actualmente, se utilizan métodos de normalización más avanzados como los basados en redes generativas adversarias o GAN, que no se han llegado a desarrollar aquí.

Finalmente, mientras que en este trabajo hemos introducido los modelos de clasificación mediante el modelo discriminativo, hay métodos de clasificación mediante el modelo generativo ampliamente utilizados en la actualidad. Uno de ellos son los *autoencoders* variacionales o VAE, basados en inferencia variacional y modelos de *deep learning*.

A. Distribuciones de probabilidad

A.1. Distribución Bernoulli y binomial

Definición A.1. Dada una variable aletatoria X, se dice que sigue una distribución Bernoulli $B(1,p)$ si describe un experimento aleatorio en el que la probabilidad de éxito es p y la de fracaso $1-p$. En dicho caso, su función masa de probabilidad es:

$$f(x) = p^x(1-p)^{1-x}, \forall x \in \{0,1\}.$$

Definición A.2. Dada una variable aleatoria X, se dice que sigue una distribución binomial $B(n,p)$ si describe n experimentos aleatorios siguiendo una Bernoulli $B(1,p)$. En dicho caso, la función masa de probabilidad de X es:

$$f(x) = \binom{n}{x} p^x(1-p)^{n-x}, \forall x \in \{1,2,\dots,n\}.$$

Proposición A.1. *Dada una distribución* $X \rightsquigarrow B(n,p)$:

- $\mathbb{E}[X] = np$
- $Var[X] = np(1-p)$

A.2. Distribución categórica

Definición A.3. Dado un vector aleatorio $\mathbf{X} = (X_1,\dots,X_n)$, se dice que sigue una distribución categórica $\text{Cat}(p_1,\dots,p_n)$ si $\sum_{i=1}^n p_i = 1$ y describe un experimento aleatorio en el que la probabilidad de pertenecer a la categoría i-ésima es p_i. Su función masa de probabilidad es:

$$f(x_1,\dots,x_k) = \prod_{i=1}^n p_i^{x_i}, \qquad \forall (x_1,\dots,x_n) : \exists i \in \{1,\dots,n\}, x_i = 1, x_j = 0 \text{ si } j \neq i$$

Proposición A.2. *Dado un vector aleatorio* $\mathbf{X} \rightsquigarrow Cat(p_1,\dots,p_n)$*:*

- $\mathbb{E}[X_i] = p_i$
- $Var[X_i] = p_i(1-p_i)$
- $Cov[X_i, X_j] = -p_i p_j$

A.3. Distribución normal univariante

Debido a la abundancia de sus propiedades, una de las distribuciones protagonistas en Estadística es la distribución normal o gaussiana.

Definición A.4. Dada una variable aleatoria X, se dice que sigue una distribución normal $\mathcal{N}(\mu, \sigma)$ si la función de densidad de probabilidad de X es:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\left(\frac{x-\mu}{\sigma}\right)^2\right), \quad \forall x \in \mathbb{R}.$$

Proposición A.3. *Dada una distribución* $X \rightsquigarrow \mathcal{N}(\mu, \sigma)$:

- $\mathbb{E}[X] = \mu$,
- $Var[X] = \sigma^2$,
- $Mo[X] = \mathbb{E}[X]$.

A.4. Distribución normal multivariante

La distribución normal univariante también posee una extensión a mayor dimensión. A partir de esta, podíamos definir el vector aleatorio con distribución conjunta normal multivariante como sigue:

- Sea $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^T$.
- Sea $\boldsymbol{\Sigma} \in \mathcal{M}_n(\mathbb{R})$ una matriz semidefinida positiva.
- Sea $\mathbf{X} = (X_1, \ldots, X_n)$ un vector aleatorio.
- Sea $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$.

Definición A.5. Decimos que $\mathbf{X} \rightsquigarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ cuando

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}{2}\right).$$

Proposición A.4. *Dada una distribución* $\mathbf{X} \rightsquigarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$:

- $\mathbb{E}[X_i] = \mu_i$,
- $Var[X_i] = (\boldsymbol{\Sigma})_{i,i}$,
- $Cov[X_i, X_j] = (\boldsymbol{\Sigma})_{i,j}$.

En dicho caso, también las distribuciones marginales y condicionadas son distribuciones normales.

Proposición A.5. *Dada una distribución* $\mathbf{X} \rightsquigarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, *y sean* $\mathbf{X}_1, \mathbf{X}_2$ *dos subvectores de manera que* $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$. *Sea* $\mathbb{E}[\mathbf{X}_1] = \boldsymbol{\mu}_1$ *y* $\mathbb{E}[\mathbf{X}_2] = \boldsymbol{\mu}_2$, *y la matriz de covarianzas:*

$$\boldsymbol{\Sigma} = \left(\begin{array}{c|c} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \hline \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{array}\right).$$

donde $\boldsymbol{\Sigma}_{11}, \boldsymbol{\Sigma}_{12}, \boldsymbol{\Sigma}_{21}, \boldsymbol{\Sigma}_{22}$ *son las submatrices resultantes de dividir* $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$. *Entonces:*

$$\mathbf{X}_1|\mathbf{X}_2 = \mathbf{x}_2 \rightsquigarrow \mathcal{N}(\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}).$$

A.5. Distribución beta

Definición A.6. Dada una variable aleatoria X, se dice que sigue una distribución beta $\beta(u, v)$ si función de densidad de probabilidad de X es:

$$f(x) = \frac{x^{u-1}(1-x)^{v-1}}{\beta(u,v)}, \quad \text{donde} \quad \beta(u,v) = \frac{\Gamma(u)\Gamma(v)}{\Gamma(u+v)}, \forall x \in [0,1].$$

Proposición A.6. *Dada una distribución* $X \rightsquigarrow \beta(u, v)$:

- $\mathbb{E}[X] = \frac{u}{u+v}$,
- $Var[X] = \frac{uv}{(u+v+1)(u+v)^2}$,
- $Mo[X] = \frac{u-1}{u+v-2}$.

A.6. Distribución Dirichlet

La distribución beta también posee una extensión a mayor dimensión. A partir de esta, podíamos definir el vector aleatorio con distribución conjunta Dirichlet como sigue:

- Sea $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$.
- Sea $\mathbf{X} = (X_1, \ldots, X_n)$ un vector aleatorio.
- Sea $\mathbf{x} = (x_1, \ldots, x_n) \in [0,1]^n$ con $\sum_{i=1}^n x_i = 1$.

Definición A.7. Decimos que $\mathbf{X} \rightsquigarrow \text{Dir}(\alpha_1, \ldots, \alpha_n)$ cuando :

$$f(\mathbf{x}) = \frac{1}{\beta(\boldsymbol{\alpha})} \prod_{i=1}^{n} x_i^{\alpha_i - 1} \quad \text{donde} \quad \beta(\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_1)\Gamma(\alpha_2)\ldots\Gamma(\alpha_n)}{\Gamma(\alpha_1 + \alpha_2 + \cdots + \alpha_n)}.$$

Proposición A.7. *Dada una distribución* $X \rightsquigarrow Dir(u, v)$ *entonces* $X \rightsquigarrow \beta(u, v)$.

B. Códigos realizados

B.1. Código R

B.1.1. Figura 1.1

```
Clases=as.factor(c(rep('c1',15),rep('c2',45),rep('c3',240)))
m1<-ggplot(data=data.frame(Clases),aes(Clases,color=Clases))+
        geom_bar(stat="count")+
        geom_text(aes(label=..count..),stat="count",position=
            position_stack(vjust=0.5));

Clases=as.factor(c(rep('c1',95),rep('c2',102),rep('c3',103)))
m2<-ggplot(data=data.frame(Clases),aes(Clases,color=Clases))+
        geom_bar(stat="count")+
        geom_text(aes(label=..count..),stat="count",position=
            position_stack(vjust=0.5));

ggarrange(m1,m2,labels=c('Muestra 1','Muestra 2'),ncol=2,nrow=1)
```

B.1.2. Figura 1.2

```
p<-seq(0,1,length=300)
prev<-par(mfrow=c(2,2))



plot(p,dbeta(p,1,1),type="l",main="u=1, v=1",xlab="",ylab="")
plot(p,dbeta(p,2,2),type="l",main="u=2, v=2",xlab="",ylab="")
plot(p,dbeta(p,3,3),type="l",main="u=3, v=3",xlab="",ylab="")
plot(p,dbeta(p,3,2),type="l",main="u=3, v=2",xlab="",ylab="")

par(prev)
```

B.2. Código python

B.2.1. Figura 1.3

```
from numpy import *
def f(mu=0,sigma=1):

        return log(1/sigma)+(sigma**2+mu**2)/2-1/2

mu=list(linspace(-2,2,500))
sigma=list(linspace(1e-5,2,500))
z=zeros((500,500))
for i in mu:
    for j in sigma:
        z[mu.index(i)][sigma.index(j)]=f(i,j)
z1=zeros((500,1))
for i in sigma:
    z1[sigma.index(i)]=f(0,i)
z2=zeros((500,1))
for i in mu:
    z2[mu.index(i)]=f(i,1)

from matplotlib.pyplot import *

figure(figsize=(18,6))


subplot(1,3,1);
plot(sigma,z1)
grid()

xlabel('Varianza')
ylabel('KL(p||q)')
axes = gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)

subplot(1,3,2);
plot(mu,z2)
xlabel('Media')
ylabel('KL(p||q)')
grid()

axes = gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)

subplot(1,3,3);
contour(mu,sigma,z,30)
```

```
scatter(0,1)
xlabel('Media')
ylabel('Varianza')
grid()

axes = gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)

savefig('kldivergencia.png')
show()
```

B.2.2. Figura 2.1

```
import pandas as pd
import seaborn as sns

from sklearn.datasets import load_iris

iris = load_iris()

iris_df = pd.DataFrame(data= iris.data, columns= iris.feature_names)
objetivo_df = pd.DataFrame(data= iris.target, columns= ['species'])
def convertir(especie):
    if especie == 0:
        return 'setosa'
    elif especie == 1:
        return 'versicolor'
    else:
        return 'virginica'
objetivo_df['species'] = objetivo_df['species'].apply(convertir)

iris_df = pd.concat([iris_df, objetivo_df], axis= 1)


import numpy as np
import matplotlib.pyplot as pl

beta=(np.linalg.inv(np.transpose(X)*X))*np.transpose(X)*Y
m=beta[1].item()
n=beta[0].item()
sigma=np.sqrt(np.linalg.norm(Y-X*beta)**2/len(Y))

pl.figure(figsize=(10,10))
valmin=min(ft).item()
valmax=max(ft).item()
```

```
x=np.linspace(valmin,valmax,500)
pl.plot(x,m*x+n,label='EMV')
pl.scatter(list(ft),list(Y),alpha=0.5,color='purple')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='blue',
    alpha=0.2,label='Varianza')
pl.grid()
pl.legend()
pl.xlabel('Longitud del s\'epalo (cm)')
pl.ylabel('Anchura del p\'etalo (cm)')
axes = plt.gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)


pl.savefig('regresionemv.png')
```

B.2.3. Figura 2.2

```
from scipy.stats import multivariate_normal
import scipy.stats

cov = [(1, 0), (0, 1)]
n=1000

x, y = np.random.multivariate_normal([0, 0], cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
p.plot_joint(sns.kdeplot, s=50, alpha=.5)

pl.xlim(-4,4)
pl.ylim(-4,4)
pl.grid()
p.plot_marginals(sns.kdeplot)


pl.savefig('regrpriori1.png')
```

```
cov = [(3, 0), (0, 0.1)]
n=1000

x, y = np.random.multivariate_normal([0, 0], cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
pl.xlim(-4,4)
pl.ylim(-4,4)
```

```
p.plot_joint(sns.kdeplot, s=50, alpha=.5)
pl.xlim(-4,4)
pl.ylim(-4,4)
pl.grid()
p.plot_marginals(sns.kdeplot)

pl.savefig('regrpriori2.png')
```

```
cov = [(0.1, 0), (0, 3)]
n=1000

x, y = np.random.multivariate_normal([0, 0], cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')

p.plot_joint(sns.kdeplot, s=50, alpha=.5)
pl.xlim(-4,4)
pl.ylim(-4,4)
pl.grid()
p.plot_marginals(sns.kdeplot)

pl.savefig('regrpriori3.png')
```

```
cov = [(3, 0), (0, 3)]
n=1000

x, y = np.random.multivariate_normal([0, 0], cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
p.plot_joint(sns.kdeplot, s=50, alpha=.5)

pl.xlim(-4,4)
pl.ylim(-4,4)
pl.grid()
p.plot_marginals(sns.kdeplot)

pl.savefig('regrpriori4.png')
```

B.2.4. Figura 2.3

```
n=1000
sigma1=1/sigma
A=sigma1*np.transpose(X)*X+np.eye(2,2)
media=sigma1*np.linalg.inv(A)*np.transpose(X)*Y
media=[media[0].item(),media[1].item()]
cov=np.linalg.inv(A)
```

```
x, y = np.random.multivariate_normal(media, cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')

p.plot_joint(sns.kdeplot, s=50, alpha=.5)
pl.scatter(media[0],media[1])
pl.grid()
pl.text(media[0],media[1], '({0:.2f},␣{1:.2f})'.format(media[0],
    media[1]))


pl.xlim(-4.5,-1.5)
pl.ylim(0.4,0.9)
p.plot_marginals(sns.kdeplot)

pl.savefig('regrposteriori1.png')
```

```
n=1000
sigma1=1/sigma
A=sigma1*np.transpose(X)*X+[[3,0],[0,0.1]]
media=sigma1*np.linalg.inv(A)*np.transpose(X)*Y
media=[media[0].item(),media[1].item()]
cov=np.linalg.inv(A)
x, y = np.random.multivariate_normal(media, cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
p.plot_joint(sns.kdeplot, s=50, alpha=.5)

pl.scatter(media[0],media[1])
pl.grid()
pl.text(media[0],media[1], '({0:.2f},␣{1:.2f})'.format(media[0],
    media[1]))

pl.xlim(-4.5,-1.5)
pl.ylim(0.4,0.9)
p.plot_marginals(sns.kdeplot)
pl.savefig('regrposteriori2.png')
```

```
n=1000
sigma1=1/sigma
A=sigma1*np.transpose(X)*X+[[0.1,0],[0,3]]
media=sigma1*np.linalg.inv(A)*np.transpose(X)*Y
media=[media[0].item(),media[1].item()]
cov=np.linalg.inv(A)
x, y = np.random.multivariate_normal(media, cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
```

```
p.plot_joint(sns.kdeplot, s=50, alpha=.5)

pl.scatter(media[0],media[1])
pl.grid()
pl.text(media[0],media[1], '({0:.2f},␣{1:.2f})'.format(media[0],
    media[1]))



pl.xlim(-4.5,-1.5)
pl.ylim(0.4,0.9)
p.plot_marginals(sns.kdeplot)
pl.savefig('regrposteriori3.png')
```

```
n=1000
sigma1=1/sigma
A=sigma1*np.transpose(X)*X+3*np.eye(2,2)
media=sigma1*np.linalg.inv(A)*np.transpose(X)*Y
media=[media[0].item(),media[1].item()]
cov=np.linalg.inv(A)
x, y = np.random.multivariate_normal(media, cov, n).T
df = pd.DataFrame({"betao":x,"beta1":y})
p=sns.JointGrid(data=df,x='betao',y='beta1')
p.plot_joint(sns.kdeplot, s=50, alpha=.5)
pl.xlim(-4.5,-1.5)
pl.ylim(0.4,0.9)
pl.scatter(media[0],media[1])
pl.grid()
pl.text(media[0],media[1], '({0:.2f},␣{1:.2f})'.format(media[0],
    media[1]))
p.plot_marginals(sns.kdeplot)

pl.savefig('regrposteriori4.png')
```

B.2.5. Figura 2.4

```
import numpy as np
import matplotlib.pyplot as plt

unos=np.ones([len(iris_df['petal␣length␣(cm)']),1])
beta=list([[0,0],[0,0],[0,0],[0,0],[0,0]])


ft=np.transpose(np.asmatrix(iris_df['sepal␣length␣(cm)']))
```

```
X=np.concatenate((unos,ft),axis=1)

Y=np.transpose(np.asmatrix(iris_df['petal␣width␣(cm)']))

beta[0]=(np.linalg.inv(np.transpose(X)*X))*np.transpose(X)*Y

Sigma=np.eye(2,2)
beta[1]=np.linalg.inv(np.transpose(X)*X-Sigma)*np.transpose(X)*Y
sigma=np.sqrt(np.linalg.norm(Y-X*beta[0])**2/len(Y))
sigma2=1/sigma

Sigma=np.mat([[3,0],[0,0.1]])
beta[2]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

Sigma=np.mat([[0.1,0],[0,3]])
beta[3]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

Sigma=3*np.eye(2,2)
beta[4]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

pl.figure(figsize=(10,10))
valmin=min(ft).item()
valmax=max(ft).item()
x=np.linspace(valmin,valmax,500)



pl.subplot(5,2,1)
pl.scatter(list(ft),list(Y),alpha=0.5)
m=beta[0][1].item()
n=beta[0][0].item()
pl.plot(x,m*x+n,label='EMV')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='blue',
    alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,3)
pl.scatter(list(ft),list(Y),alpha=0.5)
m=beta[1][1].item()
n=beta[1][0].item()
pl.plot(x,m*x+n,color='orange',label='MAP␣1')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='orange',
    alpha=0.2,label='Varianza')
pl.grid()
```

```
pl.legend()

pl.subplot(5,2,5)
pl.scatter(list(ft),list(Y),alpha=0.5)
m=beta[2][1].item()
n=beta[2][0].item()
pl.plot(x,m*x+n,color='g',label='MAP␣2')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='g',alpha
    =0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,7)
pl.scatter(list(ft),list(Y),alpha=0.5)
m=beta[3][1].item()
n=beta[3][0].item()
pl.plot(x,m*x+n,color='r',label='MAP␣3')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='r',alpha
    =0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,9)
pl.scatter(list(ft),list(Y),alpha=0.5)
m=beta[4][1].item()
n=beta[4][0].item()
pl.plot(x,m*x+n,color='purple',label='MAP␣4')
pl.fill_between(x, m*x+n+2*sigma, m*x+n-2*sigma, facecolor='purple',
    alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(1,2,2)
pl.scatter(list(ft),list(Y),alpha=0.5)

m=beta[0][1].item()
n=beta[0][0].item()
pl.plot(x,m*x+n,label='EMV')

m=beta[1][1].item()
n=beta[1][0].item()
pl.plot(x,m*x+n,color='orange',label='MAP␣1')

m=beta[2][1].item()
n=beta[2][0].item()
pl.plot(x,m*x+n,color='g',label='MAP␣2')

m=beta[3][1].item()
```

```
n=beta[3][0].item()
pl.plot(x,m*x+n,color='r',label='MAP 3')

m=beta[4][1].item()
n=beta[4][0].item()
pl.plot(x,m*x+n,color='purple',label='MAP 4')
pl.xlabel('Longitud del s\'epalo (cm)')
pl.ylabel('Anchura del p\'etalo (cm)')
axes = plt.gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)
pl.ylim(0,4)
pl.grid()
pl.legend()

pl.savefig('regresiones.png')
```

B.2.6. Figura 2.5

```
import numpy as np
import matplotlib.pyplot as pl

unos=np.ones([len(iris_df['petal length (cm)'])],1])
beta=list([[0,0,0],[0,0,0],[0,0,0],[0,0,0],[0,0,0]])


ft=np.transpose(np.asmatrix(iris_df['sepal length (cm)']))

X=np.concatenate((unos,ft,np.power(2,ft)),axis=1)

Y=np.transpose(np.asmatrix(iris_df['petal width (cm)']))

beta[0]=(np.linalg.inv(np.transpose(X)*X))*np.transpose(X)*Y
sigma=np.sqrt(np.linalg.norm(Y-X*beta[0])**2/len(Y))
sigma2=1/sigma

Sigma=np.eye(3,3)
beta[1]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

Sigma=np.mat([[3,0,0],[0,0.1,0],[0,0,1]])
beta[2]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y
```

```
Sigma=np.mat([[0.1,0,0],[0,3,0],[0,0,1]])
beta[3]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

Sigma=3*np.eye(3,3)
beta[4]=sigma2*np.linalg.inv(sigma2*np.transpose(X)*X-Sigma)*np.
    transpose(X)*Y

pl.figure(figsize=(10,10))
valmin=min(ft).item()
valmax=max(ft).item()
x=np.linspace(valmin,valmax,500)



pl.subplot(5,2,1)
pl.scatter(list(ft),list(Y),alpha=0.5)
a=beta[0][0].item()
b=beta[0][1].item()
c=beta[0][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),label='EMV')
pl.fill_between(x, a+b*x+c*np.power(2,x)+2*sigma, a+b*x+c*np.power
    (2,x)-2*sigma, facecolor='blue',alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,3)
pl.scatter(list(ft),list(Y),alpha=0.5)
a=beta[1][0].item()
b=beta[1][1].item()
c=beta[1][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='orange',label='MAP␣1')
pl.fill_between(x, a+b*x+c*np.power(2,x)+2*sigma, a+b*x+c*np.power
    (2,x)-2*sigma, facecolor='orange',alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,5)
pl.scatter(list(ft),list(Y),alpha=0.5)
a=beta[2][0].item()
b=beta[2][1].item()
c=beta[2][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='g',label='MAP␣2')
pl.fill_between(x, a+b*x+c*np.power(2,x)+2*sigma, a+b*x+c*np.power
    (2,x)-2*sigma, facecolor='g',alpha=0.2,label='Varianza')
pl.grid()
pl.legend()
```

```
pl.subplot(5,2,7)
pl.scatter(list(ft),list(Y),alpha=0.5)
a=beta[3][0].item()
b=beta[3][1].item()
c=beta[3][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='r',label='MAP␣3')
pl.fill_between(x, a+b*x+c*np.power(2,x)+2*sigma, a+b*x+c*np.power
    (2,x)-2*sigma, facecolor='r',alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(5,2,9)
pl.scatter(list(ft),list(Y),alpha=0.5)
a=beta[4][0].item()
b=beta[4][1].item()
c=beta[4][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='purple',label='MAP␣4')
pl.fill_between(x, a+b*x+c*np.power(2,x)+2*sigma, a+b*x+c*np.power
    (2,x)-2*sigma, facecolor='purple',alpha=0.2,label='Varianza')
pl.grid()
pl.legend()

pl.subplot(1,2,2)
pl.scatter(list(ft),list(Y),alpha=0.5)

a=beta[0][0].item()
b=beta[0][1].item()
c=beta[0][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),label='EMV')

a=beta[1][0].item()
b=beta[1][1].item()
c=beta[1][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='orange',label='MAP␣1')

a=beta[2][0].item()
b=beta[2][1].item()
c=beta[2][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='g',label='MAP␣2')

a=beta[3][0].item()
b=beta[3][1].item()
c=beta[3][2].item()
pl.plot(x,a+b*x+c*np.power(2,x),color='r',label='MAP␣3')

a=beta[4][0].item()
b=beta[4][1].item()
c=beta[4][2].item()
```

```
pl.plot(x,a+b*x+c*np.power(2,x),color='purple',label='MAP 4')
pl.ylim(0,4)
pl.xlabel('Longitud del s\'epalo (cm)')
pl.ylabel('Anchura del p\'etalo (cm)')
axes = plt.gca()

axes.xaxis.label.set_size(15)
axes.yaxis.label.set_size(15)
pl.grid()
pl.legend()

pl.savefig('regresionescuad.png')
```

B.2.7. Figura 2.6

```
def k(x,sigma,l):
    return(sigma*np.exp(-np.power(x,2)/(2*l)))

x=np.linspace(-10,10,500);
sigma=1
l=1
pl.figure(figsize=(7,7))
sigma=0.5
l=0.1
pl.plot(x,k(x,sigma,l),label=r"$\sigma^2=0.5,\ell^2=0.1$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.3,
    label="Regi\'on de similaridad")
sigma=0.5
l=1
pl.plot(x,k(x,sigma,l),label=r"$\sigma^2=0.5,\ell^2=1$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.2,
    label="Regi\'on de similaridad")
sigma=0.5
l=10
pl.plot(x,k(x,sigma,l),label=r"$\sigma^2=0.5,\ell^2=10$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.1,
    label="Regi\'on de similaridad")
pl.xlabel('x')
pl.ylabel('y')
pl.grid()
pl.legend()
pl.savefig('rbf1.png')
```

```
x=np.linspace(-10,10,500);

pl.figure(figsize=(7,7))
```

```
sigma=100
l=0.1

pl.plot(x,sigma*np.exp(-np.power(x,2)/(2*l)),label=r"$\sigma^2=100,\
    ell^2=0.1$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.3,
    label="Regi\'on de similaridad")
sigma=100
l=1
pl.plot(x,sigma*np.exp(-np.power(x,2)/(2*l)),label=r"$\sigma^2=100,\
    ell^2=1$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.2,
    label="Regi\'on de similaridad")
sigma=100
l=10
pl.plot(x,sigma*np.exp(-np.power(x,2)/(2*l)),label=r"$\sigma^2=100,\
    ell^2=10$")
pl.fill_between(x,0,k(x,sigma,l),k(x,sigma,l)>0.1*sigma,alpha=0.1,
    label="Regi\'on de similaridad")


pl.xlabel('x')
pl.ylabel('y')
pl.grid()
pl.legend()
pl.savefig('rbf2.png')
```

B.2.8. Figura 2.7

```
from sklearn.gaussian_process import GaussianProcessRegressor
from sklearn.gaussian_process.kernels import RBF
from sklearn.model_selection import train_test_split
from sklearn import datasets
import random
random.seed(10)

iris = datasets.load_iris()
X = iris.data[:,0]
y = iris.data[:,3]


X=np.matrix(X).T
y=np.matrix(y).T
beta=(np.linalg.inv(np.transpose(X)*X))*np.transpose(X)*y
sigma=np.sqrt(np.linalg.norm(Y-X*beta)**2/len(y))
```

```
gp = GaussianProcessRegressor(kernel=1 * RBF(), alpha = np.zeros([1,
    len(X)])+sigma)

gp.fit(X.reshape(-1,1), y.reshape(-1,1))

X_pred = np.linspace(X.min(),X.max(),500)
y_pred = gp.predict(X_pred.reshape(-1,1))

plt.figure(figsize=(10,10))
plt.scatter(list(X),list(y),color='purple',alpha=0.5)
plt.plot(X_pred,gp.predict(X_pred.reshape(-1,1)),'blue',label="
    Regresion␣GP")
plt.fill_between(X_pred, np.squeeze(gp.predict(X_pred.reshape(-1,1))
    +2*sigma), np.squeeze(gp.predict(X_pred.reshape(-1,1))-2*sigma),
     facecolor='blue',alpha=0.2,label='Varianza')

plt.xlabel('Longitud␣del␣s\'epalo (cm)')
plt.ylabel('Anchura del p\'etalo␣(cm)')
axes = plt.gca()

axes.xaxis.label.set_size(25)
axes.yaxis.label.set_size(25)

pl.grid()
pl.legend()
plt.savefig('gpregresion1.jpg')
print(gp.kernel_)
```

B.2.9. Figura 2.8

```
import matplotlib.pyplot as pl
import numpy as np

def f(x=1):
    return(1./(1+np.exp(-x)))

x=np.linspace(-5,5,500);

pl.figure(figsize=(10,10))
pl.plot(x,f(x))
pl.xlabel('x')
pl.ylabel('y')
pl.grid()
pl.savefig('logit.png')
```

```
from scipy.stats import norm
```

```
pl.figure(figsize=(10,10))
pl.plot(x,norm.cdf(x))
pl.xlabel('x')
pl.ylabel('y')
pl.grid()
pl.savefig('probit.png')
```

B.2.10. Figura 2.9

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn import datasets


iris = datasets.load_iris()
X = iris.data[:,[0,3]]

Y = iris.target
Y = np.where(Y==2, 1, Y)


logreg = LogisticRegression(C=1e5)
logreg.fit(X, Y)

[x_min, x_max]x[y_min, y_max].
x_min, x_max = X[:, 0].min() - 0.5, X[:, 0].max() + 0.5
y_min, y_max = X[:, 1].min() - 0.5, X[:, 1].max() + 0.5
h = 0.02
xx, yy = np.meshgrid(np.arange(x_min, x_max, h), np.arange(y_min, y_
    max, h))

plt.figure( figsize=(10, 10))



plt.scatter(X[:, 0], X[:, 1], c=Y, edgecolors='k', cmap=plt.cm.
    Paired)
plt.xlabel('Longitud␣del␣s\'epalo (cm)')
plt.ylabel('Anchura del p\'etalo␣(cm)')
axes = plt.gca()

axes.xaxis.label.set_size(25)
axes.yaxis.label.set_size(25)

plt.xlim(xx.min(), xx.max())
```

```
plt.ylim(yy.min(), yy.max())
plt.grid()



n11 = logreg.intercept_[0]
m11, m12 = logreg.coef_.T

n1 = -n11/m12
m1 = -m11/m12



xd1 = np.array([x_min, x_max])
yd1 = m1*xd1 + n1
plt.plot(xd1, yd1, 'k', lw=1, ls='--')
plt.fill_between(xd1, yd1, y_min, color='tab:blue', alpha=0.2)
plt.fill_between(xd1, yd1, y_max, color='tab:orange', alpha=0.2)


plt.savefig('irislog.png')
```

```
Y = iris.target
Y=Y[Y!=0]
X = iris.data[50:,[0,3]]

logreg = LogisticRegression(C=1e5)
logreg.fit(X, Y)

[x_min, x_max]x[y_min, y_max].
x_min, x_max = X[:, 0].min() - 0.2, X[:, 0].max() + 0.2
y_min, y_max = X[:, 1].min() - 0.2, X[:, 1].max() + 0.2
h = 0.02
xx, yy = np.meshgrid(np.arange(x_min, x_max, h), np.arange(y_min, y_
    max, h))



plt.scatter(X[:, 0], X[:, 1], c=Y, edgecolors='k', cmap=plt.cm.
    Paired)
plt.scatter(iris.data[:49,0],iris.data[:49,3])
plt.xlabel('Longitud␣del␣s\'epalo (cm)')
plt.ylabel('Anchura del p\'etalo␣(cm)')
axes = plt.gca()

axes.xaxis.label.set_size(25)
axes.yaxis.label.set_size(25)
```

```
plt.grid()



n1 = logreg.intercept_[0]
m1, m2 = logreg.coef_.T

n = -n1/m2
m = -m1/m2

xd = np.array([x_min, x_max])
yd = m*xd + n
plt.plot(xd, yd, 'k', lw=1, ls='--')
plt.fill_between(xd, yd, y_min, color='tab:blue', alpha=0.2)
plt.fill_between(xd, yd, y_max, color='tab:orange', alpha=0.2)

plt.savefig('irislog1.png')
```

```
iris = datasets.load_iris()
X = iris.data[:,[0,3]]
Y = iris.target

x_min, x_max = X[:, 0].min() - .5, X[:, 0].max() + .5
y_min, y_max = X[:, 1].min() - .5, X[:, 1].max() + .5
plt.figure( figsize=(10, 10))
xd1,xd= np.array([x_min,x_max]),np.array([x_min,x_max])
plt.plot(xd1, yd1, 'k', lw=1, ls='--')
plt.plot(xd, yd, 'k', lw=1, ls='--')
plt.scatter(X[:, 0], X[:, 1], c=Y, edgecolors='k', cmap=plt.cm.
    Paired)
plt.xlabel('Longitud␣del␣s\'epalo (cm)')
plt.ylabel('Anchura del p\'etalo␣(cm)')
axes = plt.gca()

axes.xaxis.label.set_size(25)
axes.yaxis.label.set_size(25)
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)
plt.grid()

plt.fill_between(xd1, yd1, y_min, color='tab:blue', alpha=0.2)
plt.fill_between(xd1, yd1, yd, color='yellow', alpha=0.2)
plt.fill_between(xd, yd1, y_max, color='tab:orange', alpha=0.2)
plt.savefig('irislogcomp.png')
```

B.2.11. Figura 3.1, Figura 3.3, Figura 3.4, Figura 3.5 y Figura 3.6

```
import numpy as np
import cv2
import os

def leer(sn,tn):
    s = cv2.imread('origen/'+sn+'.jpg')
    s = cv2.cvtColor(s,cv2.COLOR_BGR2LAB)
    t = cv2.imread('objetivo/'+tn+'.jpg')
    t = cv2.cvtColor(t,cv2.COLOR_BGR2LAB)
    return s, t

def media_sigma(x):
    x_media, x_sigma = cv2.meanStdDev(x)
    x_media = np.hstack(np.around(x_media,2))
    x_sigma = np.hstack(np.around(x_sigma,2))
    return x_media, x_sigma

def color_transfer():
    origen = ['s1','s2','s3','s4','s5']
    objetivo = ['t1','t2','t3','t4','t5']
        s, t = leer(origen[n],objetivo[n])
        s_media, s_sigma = media_sigma(s)
        t_media, t_sigma = media_sigma(t)

        altura, ancho, canal = s.shape
        for i in range(0,altura):
            for j in range(0,ancho):
                for k in range(0,canal):
                    x = s[i,j,k]
                    x = ((x-s_media[k])*(t_sigma[k]/s_sigma[k]))+t_
                        media[k]
                    x = round(x)
                    x = 0 if x<0 else x
                    x = 255 if x>255 else x
                    s[i,j,k] = x

        s = cv2.cvtColor(s,cv2.COLOR_LAB2BGR)
        cv2.imwrite('resultado/r'+str(n+1)+'.jpg',s)

color_transfer()
os.system("pause")
```

B.2.12. Figura 3.2

```
s = cv2.imread('origen/logo.jpeg')
s = cv2.cvtColor(s,cv2.COLOR_BGR2XYZ)
cv2.imwrite('resultado/logo1.jpg',s)
```

```
s = cv2.imread('origen/logo.jpeg')
s = cv2.cvtColor(s,cv2.COLOR_BGR2LAB)
cv2.imwrite('resultado/logo2.jpg',s)
```

Bibliografía

[AEH+19] Mohamed Amgad, Habiba Elfandy, Hagar Hussein, Lamees A Atteya, Mai A T Elsebaie, Lamia S Abo Elnasr, Rokia A Sakr, Hazem S E Salem, Ahmed F Ismail, Anas M Saad, Joumana Ahmed, Maha A T Elsebaie, Mustafijur Rahman, Inas A Ruhban, Nada M Elgazar, Yahya Alagha, Mohamed H Osman, Ahmed M Alhusseiny, Mariam M Khalaf, Abo-Alela F Younes, Ali Abdulkarim, Duaa M Younes, Ahmed M Gadallah, Ahmad M Elkashash, Salma Y Fala, Basma M Zaki, Jonathan Beezley, Deepak R Chittajallu, David Manthey, David A Gutman, and Lee A D Cooper. Structured crowdsourcing enables convolutional segmentation of histology images. *Bioinformatics*, 35(18):3461–3467, 02 2019. 47

[Bis06] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer-Verlag New York, Inc., 2006. xi, xiii

[Bol07] William M. Bolstad. *Introduction to Bayesian Statistics*. John Wiley and Sons, Inc., Hoboken, New Jersey, 2007. xi, xiii

[Bro13] P. Bromiley. Products and convolutions of gaussian probability density functions. 2013. 23, 47

[CL18] Kai-Seng Chou and Yan-Lung Li. Elementary inequalities. https://www.math.cuhk.edu.hk/course_builder/1819/math3060/Elementary%20Inequalities.pdf, 2018. consultado el 18 de mayo de 2021. 9

[DPA] Digital pathology asociation slide imaging repository. https://digitalpathologyassociation.org/whole-slide-imaging-repository. consultado el 21 de mayo de 2021. xi, xiii

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org. 48

[Gé17] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Sebastopol, CA, 2017. xi, xiii, 48

[KF09] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and techniques (Adaptive computation and machine learning)*. The MIT Press, Cambridge, Massachussets, London, England, 2009. xi, xiii

[LPAMÁ+21] Miguel López-Pérez, Mohamed Amgad, Pablo Morales-Álvarez, Pablo Ruiz, Lee A. D. Cooper, Rafael Molina, and Aggelos K. Katsaggelos. Learning from crowds in digital pathology using scalable variational gaussian processes. *Scientific Reports*, 11(1):11612, Jun 2021. xi, xiii, 47, 49, 50, 51

[MARC+20] Pablo Morales-Alvarez, Pablo Ruiz, Scotty Coughlin, Rafael Molina Soriano, and Aggelos Katsaggelos. Scalable variational gaussian processes for crowdsourcing: Glitch detection in ligo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. xi, xiii, 42, 45, 46

[MKB17] Jon D. McAuliffe, Alp Kucukelbir, and David M. Blei. Variational inference: A review for statisticians, arxiv:1601.00670 [stat.co]. *Journal of the American Statistical Association*, 112(518):1–9, 2017. 16

[Pri12] Simon J.D. Prince. *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012. xi, xiii

[RAGS01] Erik Reinhard, Michael Ashikhmin, Bruce Gooch, and Peter Shirley. Color transfer between images. *IEEE Comput. Graph. Appl.*, 21(5):34–41, September 2001. xi, xiii, 37, 38

[RCC98] Daniel L. Ruderman, Thomas W. Cronin, and Chuan-Chin Chiao. Statistics of cone responses to natural images: implications for visual coding. *J. Opt. Soc. Am. A*, 15(8):2036–2045, Aug 1998. 38

[RW06] Carl Edward Rasmussen and Christopher K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT press, 2006. xi, xiii, 26, 28, 30, 34

[Was04] Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer, New York, 2004. xi, xiii