Universidad de Granada

E.T.S de Ingenierías Informática y de Telecomunicación

UNIVERSIDAD
DE GRANADA

Departamento de Ciencias de la Computación e Inteligencia Artificial

Programa de Doctorado en Tecnologías de la Información y la Comunicación

Presented to obtain the degree of

Doctor of Philosophy

Probabilistic Methods for Image and Signal Classification. Applications to Medicine and Volcanology.

presented by
Miguel López Pérez

supervised by
Rafael Molina Soriano and Aggelos K. Katsaggelos

Editor: Universidad de Granada. Tesis Doctorales
Autor: Miguel López Pérez
ISBN: 978-84-1117-474-9
URI: https://hdl.handle.net/10481/76832

Agradecimientos (Acknowledgments)

Muchas gracias a todos los que me han ayudado, soportado, acompañado y/o apoyado a lo largo del camino de la tesis. Sin vosotros no habria sido así. Lo mejor de la tesis son los amigos que he conocido por el camino.

Primero agradecerle a mi director Rafael Molina. Por su esfuerzo, paciencia e ideas han sido clave no solo para la finalización de la tesis si no para mi desarollo académico, profesional y personal. También a todo el grupo de investigación, Javier, Miguel, Nicolás, Santiago, Juanga, Pablo (PHK), Fernando, Arne. Colaborar y trabajar con vosotros ha sido esencial también. Al final, todo es un trabajo en equipo. Siempre nos quedará el chino David y la Herradura. También a todos los que nos han visitado, Neel, Shuowen. Algún día hablaré vuestros idiomas. No solo investigadoras de mi grupo, también Valery, Luz, Carmen, ha sido un placer trabajar con vosotras. He aprendido mucho de vosotras.

Also thank you to my co-advisor Professor A. K. Katsaggelos and professor L. Cooper. For your welcome to USA and your guidance in digital pathology. Thanks to both of you, I completed the international adventure with interesting insights. Also, to my NU colleagues. Specially to Semih. He is totally a kardeşim. Also to Yunan. All of you made me feel like in home.

Quiero aprovechar estas líneas para agradecer también a mi familia, mi madre, mi abuela, mi tía. Por apoyarme desde siempre. A los que no están también, me haría ilusión que vieran esta hazaña que nunca hubieran pensado.

Al DB3 goloso por las tardes de trenes, dardos y montañas. Por nuestra habilidad también para resolver enigmas y por la scape room que nunca hicimos. Especialmente Andrea por ayudarme y apoyarme como la que más.

A mis amigos de Algeciras de toda la vida, a pesar de los años todo sigue igual, sois un gran apoyo. Tenemos que celebrar esto con un islita. También a los del grado en matemáticas, los panas son para siempre, y a todos los que he conocido en Granada. A los fluxions por ser el proyecto en el que confío. La ciencia puede ser divertida gracias a ellos. También, a Pato y Tomé por ayudarme a mantener la cordura en el último año de tesis gracias a la impro.

Abstract

Probabilistic methods have achieved empirical success in many predictive modeling and inference tasks. Prominent among probabilistic classifiers are Gaussian Processes (GPs). They are popular because of their expressiveness and the possibility of introducing prior beliefs. They use (probabilistic) uncertainty in modeling and inference. However, GPs can not easily estimate complex functions with stationary kernels. To overcome this limitation, Deep Gaussian Processes (DGPs) arise as their hierarchical extension. They combine the complexity of deep models while retaining the advantages of GPs.

Many areas of study take advantage of these models to solve decision-making problems. This thesis proposes and studies probabilistic methods based on GPs and DGPs for classification problems which range from supervised to weakly supervised ones. The studied models cover realistic annotation scenarios: an expert provides labels for all samples, an expert provides only label for bags of samples, and finally, multiple expert and non-expert participants provide annotations, which may not agree. The data utilized in this thesis come from: volcanology, where we have access to fully annotated data sets, histology, where to alleviate the annotation task, several medical students are asked to annotate the images and computerized tomography, where annotations are provided at scan but not slide level. We find that probabilistic models based on GPs and DGPs outperform state-of-the-art Deep Learning models for these problems.

Resumen

Los métodos probabilísticos han logrado un gran éxito experimental en muchas tareas de modelado predictivo. Entre los clasificadores probabilísticos destacan los Procesos Gaussianos (GP), los cuales son populares por su expresividad, la posibilidad de introducir conocimiento previo y utilizar la estimación de la incertidumbre en el modelado y la inferencia. Sin embargo, los GPs no pueden estimar fácilmente funciones complejas con núcleos estacionarios. Para superar esta limitación, surgen los Procesos Gaussianos Profundos (DGP) como su extensión jerárquica, combinando la complejidad de los modelos profundos a la vez que conservan las ventajas de los GPs.

Muchas áreas de estudio pueden aprovechar estos modelos para afrontar problemas de toma de decisiones. Esta tesis propone y estudia métodos probabilísticos basados en GPs y DGPs para problemas de clasificación que van desde los supervisados hasta los débilmente supervisados. Los modelos estudiados cubren escenarios realistas de anotación: un experto proporciona etiquetas para todas las muestras (aprendizaje supervisado), un experto proporciona sólo etiquetas para bolsas de muestras (Multiple Instance Learning), y finalmente, múltiples participantes expertos y no expertos proporcionan anotaciones, las cuales pueden no coincidir (crowdsourcing). Las aplicaciones vistas en esta tesis son: vulcanología, donde hemos tenido acceso a conjuntos de datos totalmente anotados por un experto, histología, donde para aliviar la tarea de anotación se ha pedido a varios estudiantes de medicina que anoten las imágenes, y detección de hemorragias en imágenes de tomografía computarizada, donde las anotaciones se han proporcionado a nivel de escáner pero no de diapositiva. Finalmente, concluimos que los modelos probabilísticos basados en GPs y DGPs superan los resultados obtenidos por los modelos de Deep Learning en el estado del arte para estos problemas.

Resumen extendido

Introducción

Los métodos probabilísticos, además de ser de gran interés y utilidad en la comunidad de aprendizaje automático, han alcanzado un gran poder predictivo en muchas tareas de toma de decisiones (Murphy, 2022). Estos modelos son capaces de abordar tareas complejas de modelado y aprendizaje, siendo adecuados en distintos ámbitos. En este marco probabilístico, destacan por su relevancia los llamados Procesos Gaussianos (GPs). Son modelos no paramétricos con aplicaciones en regresión y clasificación (Rasmussen & Williams, 2006). Tienen un gran poder predictivo, siendo capaces de obtener un alto rendimiento en bases de datos externas. Dado que los GPs estiman la incertidumbre en el modelado y la predicción, pueden medir con precisión la confianza en el resultado. Sin embargo, sólo pueden representar un número limitado de funciones. Los Procesos Gaussianos Profundos (DGPs) superan esta falta de expresividad gracias a su estructura jerárquica (Damianou & Lawrence, 2013). Constan de varios GPs apilados, de forma que la salida de un GP es la entrada del siguiente GP. Los DGPs combinan las mejores características de los modelos profundos y los GPs, permitiendo así representar funciones más complejas a la vez que conservan las ventajas de los GPs. Por estas razones, los GPs y los DGPs se han utilizado en diferentes áreas con resultados prometedores, como puede ser la medicina (Kandemir, 2015), (Li et al., 2021), la teledetección (Svendsen, Martino, & Camps-Valls, 2020; Svendsen, Morales-Álvarez, Ruescas, Molina, & Camps-Valls, 2020), y la física (Bishnoi, Ravinder, Grover, Kodamana, & Krishnan, 2021), entre muchos otros.

Además del aprendizaje supervisado en su versión clásica, esta tesis también abarca el estudio de aproximaciones para lo que se conoce como aprendizaje débilmente supervisado. En algunas áreas de aplicación, como puede ser la medicina, la obtención de etiquetas procedentes de fuentes expertas es difícil debido al alto coste que conlleva el proceso de etiquetado. En estos casos, se necesita la supervisión débil (Zhou, 2018). En esta tesis se han introducido GPs para dos tipos de clasificación débilmente supervisada: crowdsourcing (Morales-Álvarez, Ruiz, Coughlin, Molina, & Katsaggelos, 2022) y Multiple Instance Learning (MIL) (Haußmann, Hamprecht, & Kandemir, 2017). Crowdsourcing distribuye el esfuerzo de etiquetado entre múltiples anotadores con diferentes grados de experiencia. Los métodos de crowdsourcing suelen modelar el comportamiento ruidoso e inexacto de los anotadores y la etiqueta real de los expertos. En lo que respecta a MIL, los enfoques consideran etiquetas globales. Los anotadores expertos no proporcionan una etiqueta para cada muestra, sino para un conjunto (bolsa) de ellas.

Objetivos y estructura de la tesis

Esta tesis propone y aplica clasificadores probabilísticos basados en GPs y DGPs. En concreto, se estudia su aplicación a problemas supervisados y débilmente supervisados de clasificación automática de señales sísmicas de volcanes e imágenes médicas. En el caso de las aproximaciones supervisadas, abordamos la clasificación de señales volcánicas e imágenes histopatológicas. En cuanto a las débilmente supervisadas, nos centramos en aplicar crowdsourcing en histopatología, donde múltiples estudiantes de medicina proporcionan anotaciones ruidosas, y en MIL, donde los médicos expertos sólo proporcionan etiquetas globales para escáneres de tomografía computarizada. El aprendizaje supervisado se aborda en los capítulos 2-3 y el débilmente supervisado en los capítulos 4-5.

El capítulo 2 incluye nuestra contribución a las aplicaciones en el área de la vulcanología. Abordamos el problema de la clasificación automática de las ondas volcánicas-sísmicas. Además, introducimos los GPs y DGPs a la comunidad vulcanológica y mostramos cómo estos métodos probabilísticos superan al resto de los presentados hasta ahora. Esta sección también incluye una breve e intuitiva introducción a los modelos de GPs y DGPs utilizados en el resto de la tesis.

El capítulo 3 incluye la aplicación de los GPs y DGPs a la clasificación de cáncer en imágenes histopatológicas. Introducimos el uso de GPs y DGPs en problemas de clasificación del cáncer de próstata. Además, presentamos rasgos morfológicos que codifican la información a nivel de glándula, y mostramos que combinando estas características con DGPs, podemos competir con métodos de Deep Learning (DL). También presentamos un nuevo conjunto de datos públicos de cáncer de próstata con anotaciones de expertos.

El capítulo 4 estudia el uso de GPs para el aprendizaje de crowdsourcing en patología digital. Predecimos tanto la anotación de los expertos como el comportamiento de cada anotador. Lo comparamos con otros enfoques de crowdsourcing basados en DL. Además, ilustramos con mapas de segmentación los resultados obtenidos.

El capítulo 5 propone un modelo probabilístico basado en DGPs para problemas MIL. Aplicamos este modelo combinado con una red neuronal convolucional (CNN) de atención a la detección de hemorragias intracraneales en tomografía computarizada. Este modelo es entrenado en dos fases, donde en la primera, la CNN extrae los rasgos, y en la segunda, los DGPs realizan la clasificación del escáner. Además, comparamos su comportamiento con otras aproximaciones basadas en CNNs de atención y GPs. Concluimos que la complejidad de los DGPs ayuda a tener un mejor rendimiento a nivel de escáner. En concreto, reduce considerablemente el número de falsos positivos.

Conclusiones

Esta tesis demuestra que los métodos probabilísticos basados en GPs y DGPs son capaces de superar a los métodos de DL en diferentes escenarios de etiquetado (en nuestro caso, aprendizaje supervisado y débilmente supervisado) y dominios (en nuestro caso, vulcanología y medicina). Las principales conclusiones que se extraen son las siguientes:

1. En lo que respecta a la clasificación de señales sísmicas de volcanes, los GPs y DGPs superaron a los métodos de DL funcionando mucho mejor en la detección de clases raras. Además, los GPs y DGPs estimaron mejor la incertidumbre, proporcionando probabilidades más precisas.

2. En cuanto a la detección del cáncer de próstata, demostramos que las características extraídas del espacio de densidad óptica codificaban información más relevante que las extraídas del espacio RGB. Además, las características morfológicas y de textura obtuvieron los mejores resultados al clasificarlos con GPs o DGPs. Demostramos que los GPs y DGPs superaron a cualquier otro clasificador no profundo, y también fueron competitivos con los métodos basados en DL. Finalmente, demostramos empíricamente que los GPs y DGPs son más eficientes que los métodos basados en DL.

3. En cuanto a la clasificación usando crowdsourcing en cáncer, un GP entrenado con características extraídas de una red neuronal profunda preentrenada fue capaz de obtener mejores resultados que los métodos basados en DL. Los GPs usando crowdsourcing modelaron automáticamente las etiquetas ruidosas y la experiencia de cada anotador. Este modelo, entrenado con etiquetas ruidosas, fue competitivo con el entrenado con anotaciones de expertos en la clasificación del cáncer de mama. Observamos que el crowdsourcing es una solución factible para la falta de datos etiquetados, ya que las imágenes de cáncer pueden ser anotadas masivamente por estudiantes de medicina.

4. En cuanto al uso de MIL en la detección de hemorragias cerebrales, el DGP-MIL propuesto logró mejores resultados que el basado en DL y GP no profundos. Además, demostramos la necesidad de modelos jerárquicos basados en GPs para aprender funciones complejas en aplicaciones reales. Este modelo fue capaz de obtener mejores resultados tanto a nivel de escaneo como de cortes, y su precisión fue notablemente mejor. Fue capaz de identificar con mayor precisión los falsos positivos, resultando así en un clasificador más robusto para su uso en medicina. Estos resultados abren una nueva puerta para el etiquetado eficiente y la posibili-

dad de entrenar modelos más potentes con una menor dependencia del etiquetado exhaustivo por un experto.

Contents

Chapter 1

Introduction

> The next generation of data-efficient learning approaches relies on us developing new algorithms that can propagate stochasticity or uncertainty right through the model
>
> Neil Lawrence

Probabilistic methods are of great interest and use in the machine learning community (Murphy, 2022). They can deal with difficult modeling and learning tasks, being suitable for many different scenarios. In this framework, Gaussian Processes (GPs) are very popular. GPs are fully probabilistic non-parametric models with applications in regression and classification (Rasmussen & Williams, 2006). Since GPs estimate the uncertainty in modeling and prediction, they can accurately measure the confidence on the outcome. GPs encode prior information in a kernel function, acting as a strong regularizer. In contrast to deep models, they have to learn fewer parameters to estimate a complex model. These characteristics usually lead to a better generalization capability. In brief, GPs perform well in unseen data, even when data is scarce. However, they can only represent a limited number of functions. Deep Gaussian Processes (DGPs) overcome this lack of expressiveness due to their hierarchical structure (Damianou & Lawrence, 2013). They consist of several stacked GPs, where the output of one GP is the input to the next GP. DGPs combine the best features of deep models and GPs. This hierarchical extension can represent more complex functions while retaining the advantages of GPs. For these reasons, GPs and DGPs have been used in different areas with promising results: medicine (Kandemir, 2015; Li et al., 2021), remote sensing (Svendsen, Martino, & Camps-Valls, 2020; Svendsen, Morales-Álvarez, et al., 2020), and physics (Bishnoi et al., 2021), among many others.

Together with classical supervised learning, one important topic covered in this thesis is *weakly supervised learning*. In some areas of application, such as medicine, obtaining fine-grained expert labels is difficult due to the high cost of the data labeling process.

In these cases, weak supervision is needed (Zhou, 2018). GPs have been introduced for two types of weakly supervised classification: crowdsourcing (Morales-Álvarez et al., 2022) and Multiple Instance Learning (MIL) (Haußmann et al., 2017). Crowdsourcing distributes the effort of labeling among multiple annotators with varying expertise. Crowdsourcing methods usually model the noisy annotator behavior and the expert ground truth. Regarding the MIL setting, it usually considers coarse-grained labels. Expert annotators do not provide a label for each sample but for a bag of them.

This thesis proposes and applies probabilistic classifiers based on GPs and DGPs. Specifically, we study their application to supervised and weakly supervised problems in volcano-seismic signals and medical images. Regarding the supervised ones, we address volcano-seismic signal and histopathological images classification. If we focus on the weakly supervised ones, we address crowdsourcing in histopathology, where multiple medical students provide noisy annotations, and MIL, where the expert doctors only provide global labels for whole CT scans.

The rest of the chapter is structured as follows. First, Section 1.1 provides a brief and intuitive introduction to our tool, GPs, and DGPs. We describe their mathematical formulation and include graphical examples for a deeper understanding. Then, Section 1.2 covers the supervised learning tasks addressed in this thesis. These applications range from volcanology to medicine. Then, Section 1.3 introduces the weakly supervised approaches, i.e., crowdsourcing with histopathological images and Multiple Instance Learning in CT scans. Section 1.4 collects the objectives. Section 1.5 explains the methodology used. Finally, Section 1.6 discloses the results and structure of the rest of the thesis.

1.1 Gaussian Processes

Before introducing the areas of application studied in the thesis, we provide a brief and graphical introduction to GPs and DGPs, the probabilistic classifiers studied. Also, this theory is presented in the papers included in this thesis.

A Gaussian Process prior assumes a multivariate normal distribution on the latent variable $\mathbf{f} = (f_1, ..., f_N)^\intercal$ given $\mathbf{X}$. A mean and a kernel (covariance) functions define this prior distribution. Without loss of generality, the mean function $\mu(\mathbf{x})$ is usually set to $\mathbf{0}$. The kernel $k(\mathbf{x}, \mathbf{x}')$ encodes the prior belief about the data. It encapsulates the characteristics of the functions that the GP is going to estimate. The most popular kernel is the Squared Exponential (SE). It has a great power of representation. Also, it imposes smoothness on the latent function $\mathbf{f}$. The SE kernel is defined as $k_{\mathrm{SE}}(\mathbf{x}_i, \mathbf{x}_j) = C \exp\left(\frac{-||\mathbf{x}_i - \mathbf{x}_j||^2}{2l^2}\right)$, where the parameters C and l are usually estimated by maximum likelihood.

Once we have modeled the latent function $\mathbf{f}$ using a GP prior, we have to define the observation model $\mathrm{p}(\mathbf{y}|\mathbf{f})$, where $\mathbf{y}$ is the noisy observed variable. The observation model depends on the specific task. For binary classification, the common likelihood is the Bernoulli distribution, i.e., $\mathrm{p}(y_i|f_i) = \mathrm{Ber}(y_i; \mathrm{sigmoid}(f_i))$. The joint density of $\mathbf{y}$ and $\mathbf{f}$ becomes,

$$\mathrm{p}(\mathbf{y}, \mathbf{f}) = \underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n)}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{f})}_{\text{GP prior}} \ , \tag{1.1}$$

where we assume independence across the instance labels given the latent variables. The goal becomes the estimation of the model parameters, in this case C and l, and the calculation of the (posterior distribution of) latent function given the (observed) training data $\mathrm{p}(\mathbf{f}|\mathbf{y})$.

One main drawback of Gaussian Processes is their scalability. They have a high computational cost $\mathcal{O}(N^3)$ because their formulation involves the inversion of an $N \times N$ matrix. Sparse GPs have been proposed to overcome the scalability problem (Titsias, 2009). They use $\tilde{M} \ll N$ inducing points u_m which are GP realizations at inducing locations $\mathbf{z}_m$. We can see this as $f(\mathbf{z}) = u$. The inducing points encode the information of the observations in a few points. Their locations $\{\mathbf{z}_m\}_{m=1}^{M}$ are estimated while learning. This approach lightens the computational cost to $\mathcal{O}(n\tilde{M}^2)$. However, the posterior distribution is intractable and approximate inference must be used. The Scalable Variational Gaussian Process (SVGP) inference is the state of the art for sparse GPs (Hensman, de G. Matthews, & Ghahramani, 2015). Furthermore, it allows to train in mini-batches. The joint density in this case is given by

$$\mathrm{p}(\mathbf{y}, \mathbf{f}, \mathbf{u}) = \underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n)}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{f}|\mathbf{u}; \mathbf{Z})\mathrm{p}(\mathbf{u}; \mathbf{z})}_{\text{sparse GP prior}}, \tag{1.2}$$

the semicolon notation indicates which are the deterministic inputs of each function. The goal here is to calculate $\mathrm{p}(\mathbf{u}, \mathbf{f}|\mathbf{y})$ and estimate the model parameters.

Figure 1.1 shows an example of Sparse Gaussian Processes for a 1-dimensional regression problem. In a regression problem, we observe noisy data produced by an unobserved latent function. The goal is to approximate the latent function by learning a function from the noisy data. We see that the GP mean approaches the latent function. Furthermore, the latent function is inside the confidence interval. The estimated uncertainty reflects the lack of knowledge of the model, for instance, in areas with less inducing points. Also notice that the optimal location for the inducing points is where the function has more variations. Figure 1.2 shows an example of GPs for binary classification in a 1-dimensional toy problem. In (a), we draw samples for the posterior distribution of the latent variable $\mathrm{p}(\mathbf{f}|\mathbf{y})$. Note that all the samples share the same level of smoothness.

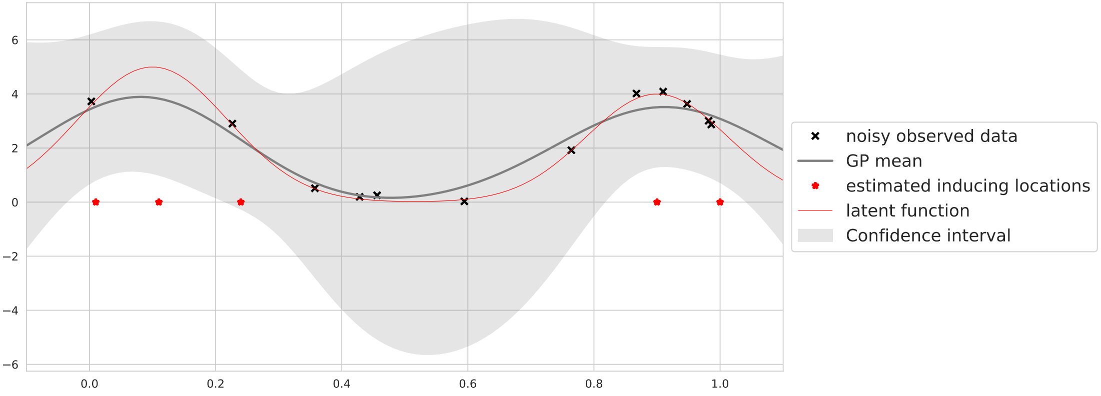
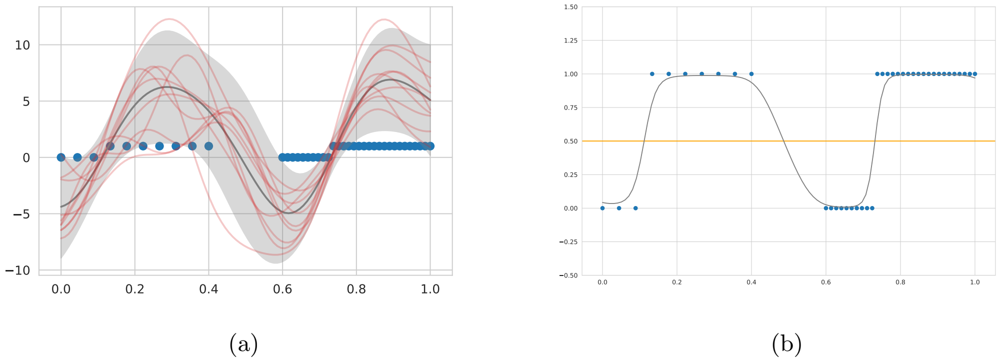


Figure 1.1: Example of a Sparse Gaussian Process on a 1-dimensional regression problem.



(a)

(b)

Figure 1.2: 1-dimensional binary classification problem with the input dimension on the x-axis and the output dimension on the y-axis. In (a) we draw the distribution of the latent function p(f_*). In (b) we draw p($y_* = 1$).

In (b), the sigmoid function squashes the latent functions to obtain the probabilities of the observed samples.

1.1.1 Deep Gaussian Processes

A DGP is a hierarchical model which consists of several stacked SVGPs (Damianou & Lawrence, 2013). We define $\{\mathbf{F}^l\}_{l=1}^{L}$ latent variables where each $\mathbf{F}^l$ follows a GP prior with input locations given by $\mathbf{F}^{l-1}$. We consider $\mathbf{F}^0 = \mathbf{X}$. We denote $f_{n,d}^l$ as the latent variable value for the n-th instance in the dimension d (being $1 \leq d \leq D^l$) for the layer l. Notice that in this problem $D^L = 1$. The vector f_n^l contains all the dimensions for the n-th instance in the l-th later. For binary classification, again, the likelihood is defined by a Bernoulli distribution,

$$\mathrm{p}(y_n|f_n^L) = \sigma(f_n^L)^{y_n}\left(1 - \sigma(f_n^L)\right)^{1-y_n}. \tag{1.3}$$

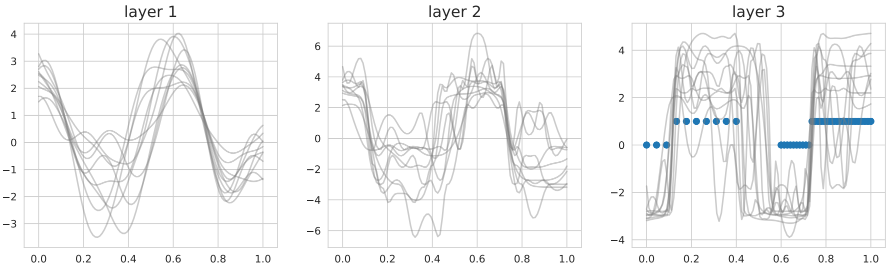


Figure 1.3: 3-layer Deep Gaussian Process for binary classification. We saw the latent representation of each hidden layer.

Assuming independence across the instance labels given the latent variables, we obtain,

$$\mathrm{p}(\mathbf{Y}|\mathbf{f}^L) = \prod_{n=1}^{N} \mathrm{p}(y_n|f_n^L). \tag{1.4}$$

Because of the computational cost, we introduce again the so called sparsity. We have M^{l-1} inducing locations $\mathbf{Z}^{l-1}$ at each layer l with inducing values $\mathbf{U}^l$ for each dimension. So we can write the joint density function,

$$\begin{aligned}\mathrm{p}(\mathbf{Y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = &\underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n^L)}_{\text{likelihood}} \\ &\times \underbrace{\prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1})}_{\text{DGP prior}}.\end{aligned} \tag{1.5}$$

Exact inference is unfeasible for DGPs. In this thesis, we follow the Doubly Stochastic Variational Inference Salimbeni and Deisenroth (2017). Furthermore, it allows to train in mini-batches.

Figure 1.3 shows samples of the latent function across the hidden layers of a DGP, from the first and second layer, which are the hidden representation features learned. Then, the third (final) layer is the output for the final classification. We can see that the first layer are actually shallow GPs. Then, when we apply a GP to these features we can obtain more complex patterns as shown in the second and third layers. The flat regions are smooth while the jumps in the decision boundaries are abrupter. Despite being a simple problem, we can see the superiority of DGPs over GPs. This fact encourages their use for complex tasks as the studied throughout this thesis and the introduced in the following sections.

1.2 Supervised learning

The data utilized for this task in this thesis come from volcanology and histopathology. This section introduces both fields and contextualizes the contributions.

1.2.1 Volcano-seismic signals

Volcanoes pose a hazard to property and population in several geographic areas. For this reason, high-risk volcanoes are monitored to avoid further damage. Stations placed near volcanoes perform this monitoring process. They capture the elastic waves produced by different phenomena. These seismic signals offer insights into the internal dynamics of the volcano. The objective is to classify seismic events from the registered signals and associate them to their original geophysical source mechanism. A good classification of these events can lead to the detection of events that precede eruptions. These events are crucial to obtain an early-warning system that can help evacuate people and save lives. Furthermore, we can also explain the dynamics of the volcano by analyzing the spotted patterns. Unfortunately, as it happens in other areas, such as medical imaging, large enough databases with high-quality labels are even more scarce. For this reason, GPs, which have never been applied to this problem before, are also suitable.

In Chapter 2, we classify events recorded at the *Volcán de Fuego de Colima*, in Colima (Mexico). It is a complex scenario where hierarchical models based on deep neural networks have been proposed. The Deep Belief Network (DBN) and a stacked denoising autoencoder (sDNA) were compared to other state-of-the-art isolated events classifiers in this database (Titos, Bueno, Garcia, & Benitez, 2018). We propose GPs and DGPs to address this problem with satisfying results. GPs and DGPs models outperformed DNNs and estimated better the uncertainty in the predictions.

1.2.2 Histopathological images

According to the World Health Organization, cancer is a leading cause of death worldwide, accounting for nearly 10 million deaths in 2020. The gold standard for cancer detection is a biopsy. An expert pathologist has to analyze the samples on a microscope, a process which is subjective and time consuming.

Digital scanners allow the storage of these samples as histological images. Then, Machine Learning (ML) methods can analyze these images. These facts encourage automatizing the diagnosis task. Computer-Aided Diagnosis (CAD) systems use ML techniques to help pathologists to provide an accurate and fast diagnosis. CAD systems decrease the workload considerably, enabling doctors to focus on problematic cases.

CAD systems in histopathology have incorporated popular DL methods successfully. For example, DL methods previously presented for object classification have been used in histopathology with satisfying results, such as VGG16, Inception v3, ResNet50, or

EfficientNet (Ferlaino et al., 2018; Koné & Boulmane, 2018; Yang et al., 2021). Transfer learning adapted this models for medical imaging (Ahmed et al., 2021; Shallu & Mehra, 2018). These deep neural networks have also been used as a backbone for segmentation models (Kim et al., 2021; Priego-Torres, Sanchez-Morillo, Fernandez-Granero, & Garcia-Rojo, 2020).

Chapter 3 includes the contributions of this thesis to histological cancer classification. We introduce GPs to the digital pathology community. We compare GPs and DGPs to DL methods. We show that GPs and DGPs are competitive to state-of-the-art DL methods and, in some cases, perform better.

1.3 Weakly supervised learning

So far we have studied the use and performance of GP and DGPs in supervised tasks. However, the use of supervised models may not be feasible in some medical imaging problems. The need for expert doctors to label data, often results in weakly labeled databases. This thesis also contributes to weakly supervised learning problems by proposing probabilistic methods based on GPs and DGPs.

1.3.1 Crowdsourcing in histopathological images

Crowdsourcing has emerged as a solution for efficient labeling. The main idea is to engage a broad set of participants to annotate the images. Typically, the degree of expertise varies among the participants. Thus, crowdsourced labeled data suffer from high labeling noise. One common approach is to fix/aggregate all the labels in a previous step, for example, majority voting. These methods tend to fail when most images have only one label. Generally, it is better to keep each annotation and model the reliability of each annotator separately (Karimi, Dou, Warfield, & Gholipour, 2020). Then, the method simultaneously estimates a latent classifier that weights the labels of each annotator with estimated reliability. Regarding this crowdsourced framework, Nir et al. (2018) applied crowdsourcing to prostate cancer grading in histopathology images. Morales-Álvarez et al. (2022) first proposed GPs for crowdsourcing (SVGPCR) to detect glitches in LIGO.

In Chapter 4, we adapt SVGPCR to breast cancer classification with promising results. We predict the ground truth from non-expert annotations. We also estimate the participants' reliability and behavior.

1.3.2 Multiple Instance Learning in CT scans

Intracranial hemorrhage (ICH) has high mortality and can produce permanent disability, thus, early diagnosis and proper treatment are essential for recovery. Computed Tomography (CT) is a non-invasive technique for ICH diagnosis. CT scans are cheap and accessible for patients and provide fast results for radiologists. A CT scan produces several images (slices) of the brain from different angles. Consequently, radiologists can misdiagnose cases due to fatigue while screening these trials Strub, Leach, Tomsick, and Vagal (2007). CAD systems can reduce the workload of radiologists and provide a fast and accurate diagnosis.

DL methods based on CNNs have been widely studied in ICH detection. The most straightforward way is to apply DL models at the slice level (Cho et al., 2019; Phong et al., 2017), which consists in training models with the labeled slices and then to predict at slice level. However, it is expensive to collect labels at the slice level. Scan labels are easy to obtain since they already appear in the clinical report. To leverage scan labels, 3D CNNs have been applied (Jnawali, Arbabshirani, Rao, & M.d, 2018; Titano et al., 2018). The main problem with these approaches is that 3D CNNs are computationally expensive. Furthermore, 3D CNN can not localize where the injury is, which is crucial for an interpretable prediction.

Multiple Instance Learning allows the easy obtention of labeled data. Since obtaining fine-grained annotations is high-time consuming, one feasible approach is using coarse ones. MIL avoids the exhaustive labeling of the image. It relies instead on global labels, which describe and diagnose the whole medical scan. This framework is specially difficult for supervised ML methods because they do not know the exact patterns of the studied disease. Commonly, MIL methods can predict local and global labels on unseen whole medical scans. The local prediction is of real utility for practical implementation. MIL has been studied in medical imaging both in histopathological images, and CT scans (Campanella et al., 2019; Wu, Schmidt, Hernández-Sánchez, Molina, & Katsaggelos, 2021).

Chapter 5 proposes a probabilistic model based on Deep Gaussian Processes for MIL. It is the first time that DGPs are proposed for this problem. We study the application of ICH detection in CT scans. As commented before, globals labels of CT scans are already in the clinical report. MIL methods overcome the limitations of 3D CNNs. They do not need a large amount of computational resources and also can predict at slice level. We predict at slice and scan level achieving excellent results compared to CNNs.

1.4 Objectives

After introducing the problems addressed in this thesis, we present the main objectives. They include the development of probabilistic methods based on GPs and DGPs and their application to different classification problems in medicine and volcanology. We consider both fully (supervised) and weakly supervised tasks. Specifically, the objectives of the thesis are as follows:

- **To develop state-of-the-art methods for automatic classification of volcano-seismic signals with Gaussian Processes and Deep Gaussian Processes.**

So far, GP-based models have not been applied in the field of volcanology. Only DL methods have been studied in this domain. We truly believe that, due to the small databases, GPs and DGPs-based models will perform better than state-of-the-art deep neural networks.

- **To develop state-of-the-art methods for supervised classification in digital pathology with Gaussian Processes and Deep Gaussian Processes.** Although Gaussian Processes have been applied with success in this field, DGPs have not ever been proposed for histopathological problems. We seek to study the behavior and performance of these probabilistic classifiers compared with state-of-the-art DL methods. We also investigate the feasibility of using both handcrafted and deep features.

- **To address crowdsourcing classification in medical imaging using Gaussian Processes.** Since labeling in this domain poses a problem, this thesis aims to study for the first time a probabilistic GP-based model with data collected from multiple non-expert participants. Here, the problem is twofold. The objective is both to discern the ground truth and how to model the noisy behavior of non-expert. Our goal is also to estimate the reliability of each participant.

- **To improve state-of-the-art Multiple Instance Learning methods for medical imaging with Gaussian Processes and Deep Gaussian Processes.** MIL provides a solution for sparse labeling. This thesis aims to tackle this problem with GPs and DGPs and study them against previous methods based on shallow GPs and DL. Our objective is not only to predict at bag level but also instance level, in other words, to localize the lesion.

1.5 Methodology

To fulfill the objectives, we present below the methodology designed for this thesis. As the study involves an exhaustive experimentation and all the objectives must be proved empirically, the methodology is close to the scientific method. The guidelines applied will be:

1. **Observation:** We first study the literature regarding GPs and DGPs as well as previous techniques used in the addressed domains.

2. **Data collection:** We collect real-world data from different sources to both train and assess the algorithms. We consider public databases whenever possible.

3. **Hypothesis formulation:** We select state-of-the-art models and propose new ones to improve the results. We address the problems presented in the objectives.

4. **Experimentation:** We perform rigorous experimentation with the collected data in step two. We use the computation resources of the Visual Information Processing research group of the University of Granada. Since the datasets addressed in this thesis are mostly imbalanced, the f1 score is of crucial importance.

5. **Hypothesis contrast:** We compare, analyze and validate the results obtained in the experimentation against the state-of-the-art techniques in the literature.

6. **Demonstration or refutation of the hypothesis:** We check if the extracted conclusions agree with the hypothesis previously formulated. If the results do not satisfy them, we will go back to step three and formulate a new hypothesis.

7. **Thesis extraction:** We formalize the conclusions during the research process and justify the developed methods through the experimentation. All the proposals and results are synthesized in this memory.

1.6 Results

This section provides the main results obtained using the methodology to pursue the objectives. The main contributions are at the beginning of the corresponding chapter, and Chapter 6 exposes the conclusions of these results and future work. In this thesis, the results can be separated into two different problems: supervised and weakly supervised learning.

1.6.1 Supervised learning

We address supervised learning using Gaussian Processes and Deep Gaussian Processes. We develop probabilistic methods based on them with application in volcanology and medical imaging.

Chapter 2 includes our contribution to volcanology in one journal article (JCR Q1). We address the problem of the automatic classification of volcano-seismic waves. We introduce GPs and DGPs to the volcanology community and show how these probabilistic methods outperform state-of-he-art DL methods presented so far. The DL methods are DBN and sDNA. We obtain not only better global results but also much better performance in imbalanced classes. Also, the predictions of the GPs are more accurate. Finally, GPs provide explainability about the importance of the different features. This work also includes a brief and intuitive introduction to GPs and DGPs used in the rest of the thesis.

Chapter 3 includes the application of GPs and DGPs to supervised cancer classification of histopathological images in one journal article (JCR Q1). We introduce GPs and DGPs to prostate cancer classification. We design handcrafted features which encode morphological information. We also obtain that by combining (morphological and texture) handcrafted features with DGPs, we fairly compete with state-of-the-art DL methods in histopathology. These methods are Inception v3, Xception and VGG19. We obtain that GPs and DGPs are computationally more efficient. Finally, we release a new public dataset of prostate cancer with expert annotations from the Hospital Clínico Universitario de Valencia.

1.6.2 Weakly supervised learning

Labeling medical images is costly and time-consuming. To alleviate the data collection process, recent techniques rely on weakly supervised learning. Here, we present the main results obtained from two different approaches: crowdsourcing and multiple instance learning.

Chapter 4 studies the use of GPs for crowdsourcing learning in digital pathology in one journal article (JCR Q1). We obtain that this probabilistic crowdsourcing model outperforms other state-of-the-art DL methods. The backbone for feature extraction used to compare and study the model is the VGG16. GPs for crowdsourcing predict well the ground truth as well as the noisy annotator behavior. The segmentation maps illustrate that this method is useful for medical image analysis. Finally, this crowdsourcing model performs closely to the one trained with expert labels. This framework enables efficient labeling of medical images by engaging non-experts, for example, medical students, instead of experts.

Chapter 5 proposes a probabilistic model based on DGPs for MIL problems in one journal article (JCR Q1). We apply this model combined with an attention CNN to intracranial hemorrhage detection in CT scans. We compare its behavior with other approaches based on attention CNNs and shallow GPs. We conclude that the complexity of DGPs helps perform better at the scan level. This approach allows to work with larger databases since the global (bag) labels are in medical reports without the need for expert doctors for data labeling. Finally, this system outperforms the precision reached by previous approaches, thus, reducing considerably the number of false positives. This result is of vital importance for practical systems in hospitals where the reliability of the predicted results is fundamental.

Chapter 2

Automatic classification of volcano-seismic events with Deep Gaussian Processes

2.1 Publication details

Authors: Miguel López-Pérez, Luz García, Carmen Benítez, Rafael Molina.
Title: A Contribution to Deep Learning Approaches for Automatic Classification of Volcano-Seismic Events: Deep Gaussian Processes.
Publication: IEEE Transactions on Geoscience and Remote Sensing, vol. 59, no. 5, 3875-3890, May 2021.
Status: Published.
Quality indices:

- Impact Factor (JCR 2020): 5.600.
- 35/273 (Q1) en Engineering, Electrical & Electronic.

2.2 Main contributions

- This work is the first paper in the literature on the use of GPs for volcano-seismic event classification.
- We elaborate an intuitive introduction for GPs and DGPs, including a graphical motivation for the use of these models.
- We conduct a comprehensive and insightful study of GPs and DGPs against other methods based on deep neural networks. GPs and DGPs outperform the rest and also predict more reliable probabilities.

A contribution to Deep Learning approaches for automatic classification of volcano-seismic events: Deep Gaussian Processes

Miguel López-Pérez[a,a], Luz García[b], Carmen Benítez[b], Rafael Molina[a]

[a] *Department of Computer Science and Artificial Intelligence, University of Granada, 18010 Granada, Spain.*

[b] *Department of Signal Theory, Telematics and Communications, University of Granada, 18071 Granada, Spain.*

Abstract

Automatic classification of volcano-seismic events is a key problem in volcanology. Due to its complexity, Deep Learning (DL) techniques have become the tool of choice for this problem, outperforming classical classifiers. The main drawback of this approach, when applied to the classification of volcano-seismic events, is its tendency to overfit because of the small-size available databases. In this work, we propose and analyze the use of Gaussian Processes (GPs) and Deep Gaussian Processes (DGPs), their hierarchical extension, for volcano-seismic event classification. We empirically prove the adequacy of the proposed modelling with an insightful and exhaustive comparison with state-of-the-art DL-based methods on a seismic database recorded at "Volcán de Fuego", in Colima (Mexico). The hierarchical structure of DGPs and the reduced number of parameters to be automatically estimated become essential to achieve an excellent performance even on small databases, capturing well the complex patterns of seismic signals for all classes and in particular for those which have been hardly observed.

Keywords: Geoscience and remote sensing, geophysical signal processing, remote monitoring, remote sensing, signal processing, volcanoes, volcanic activity, gaussian processes, deep gaussian processes, deep learning.

1. Introduction

Geophysical processes like displacements of magma and other fluids or gases, or fractures of solid materials in volcanic areas, are derived from the exchange of elastic energy between volcanic structures and their surroundings. Seismic signals registered by stations deployed near volcanoes capture elastic waves that reflect such exchanges. Their study provides very valuable information. When properly interpreted, seismic signals offer a useful insight into the internal dynamics of the volcano. Source mechanisms originating them can be inferred from their analysis, together with information about the earth's crust materials traversed during the trip of the elastic wave towards the station registering it [1][2].

[*] Corresponding author.

This work was supported by the Ministerio de Economía y Competitividad (MINECO) through projects DPI2016-77869 and A-TIC-215-UGR18, by the Ministerio de Ciencia e Innovación through projects PID2019-105142RB-C22 and PID2019-106260GB-I00, and by the European Comission (action MSCA-ETN-ITN) through project SEP-210560672.

Email address: `mlopez@decsai.ugr.es` (Miguel López-Pérez)

Detection and classification of seismic events consists of processing seismic registers to spot events and associate them with their originary geophysical source mechanism based on the characteristics of the signal. The source mechanism inference is a complex task given the number of additional factors that influence the signal arriving at the seismic station. The degree of elasticity/anisotropy of materials in the source location, distance to the station, characteristics of the propagation path, or frequency response of the registering instrument are examples of them. Once detected, the spotted patterns in the sequence of events are analyzed to understand the physical model explaining the dynamics of the volcano. They are also used in applications like early-warning monitoring systems based on the detection of events precursors of eruptions.

In the last decades, the amount of seismic data available has increased enormously together with the computing and storage capacity. These facts have encouraged the geophysical community to explore the use of Machine Learning (ML) algorithms for automatic classification of seismic events [3][4][5]. ML techniques avoid the tedious and repetitive work of manual labeling, often done by geophysical experts, and increase the capacity to process enormous volumes of data. They capture complex data correlations not detectable by human experts. There is a wide range of possible ML algorithms usable for automatic classification of seismic events. The election of the approach depends on factors like the dimensionality of the data and corresponding classes, the size of the labeled training database, the continuous/isolated classification objective needed, or the interpretability of the model searched for.

Within the field of seismicity, the classification of volcano-seismic events presents specific challenges derived from their origins. Simultaneous seismic events related to liquid and/or gas-solid processes take place in the volcanic scenario. Tremors, long period events, or surface effects like rockfalls, landslides, or pyroclastic density flows might happen simultaneously generating complex seismic registers with overlapped events. In addition, volcanic regions present changing propagation and site properties. Sismo-volcanic sources are often shallower compared to tectonic ones. As a consequence, *near-source* and *surface-propagation* effects complicate the analysis of the seismic signal. The labeling task must therefore be carried out by expert geophysicists with a deep knowledge of the particular volcano generating the data. This is a difficult, tedious, and time-consuming task that requires deep expert knowledge and a strict maintenance of the labeling criteria. For all these reasons, large enough databases with high quality labels are scarce, but extremely necessary to improve the knowledge of the volcanic structures and predict their behavior.

Supervised ML techniques for automatic classification of isolated volcano-seismic events started around 2005 with the usage of Artificial Neural Networks (ANN) in the pioneer [6]. Since then, interesting applications like [7], and models based on Support Vector Machines (SVM) [8], combination of several *shallow* classifiers like ANN and SVM [9] or ANN and Genetic Algorithms [10] have been developed. In parallel, Hidden Markov Models [11][12][13][14] have been introduced to model temporal structures, providing approaches to successfully detect and classify events in continuous seismic registers.

DL approaches, with higher degree of abstraction and knowledge extraction for complicated data sets, became popular after the proposals of Hinton in 2006 [15] and Bengio in 2012 [16] (accompanied by important advances in computational power). These two works proposed, respectively, to use Restricted Boltzman Machines (RBMs) and Denoising Autoencoders (DAs) to initialize hidden layers via unsupervised layer-by-layer training, proving that Deep Networks could be trained well, with more optimal initializations and useful learned representations of the data.

DL was first applied to image processing and speech, and has spread its usage to many disciplines, with attractive applications in the field of seismology. Examples of them are the automatic P-phase picking approach in [17], the skip connection CNN proposed in [18] to detect geyser related events in continuous registers, or the usage of Deep Convolutional Autoencoders for seismic signal clustering in [19]. Classification of volcano-seismic signals using Deep Neural Networks (DNNs) was first presented by [20] with the implementation of a Deep Belief Network (DBN) and a stacked denoising autoencoder (sDNA). Their classification performance was compared to the state-of-the-art isolated events classifiers on the seismic events database of the *Volcán de Fuego de Colima* (México). The work in [21] implemented and compared three Recurrent Neural Network (RNN) architectures (Vanilla, LSTM and GRU) to detect and classify volcano-seismic events from the *Volcán de Decepción* (Antarctica) in continuous registers. Unfortunately, the use of DL techniques is based on the availability of large amounts of data. To overcome the lack of large databases of labeled volcano-seismic events necessary for effective classification with DL architectures, Transfer Learning approaches based on DL have been explored in [22] and [23].

In the ML scenario described so far, GP models were introduced in 2006 [24]. They are non-parametric probabilistic models which deal with uncertainty in prediction and modeling. Interesting connections between DNNs and GPs were studied in [25], where the correspondence between GPs and priors for *infinitely wide* DNNs was established. Their expressiveness and robustness to overfit have been largely praised. The prior information in the kernel function of the GPs acts like a regularizer making them suitable for not very large databases, which is the case in volcanology. This is in contrast to neural networks which have to learn a huge amount of parameters to estimate a complex model and so they tend to overfit on small databases. Furthermore, as N. Lawrence indicates in his post [26], *the next generation of data efficient learning approaches relies on us developing new algorithms that can propagate stochasticity or uncertainty right through the model.* See also [27] and the seminar thesis [28].

Although GPs are very flexible, they suffer from a severe limitation. They are commonly used with stationary kernels which makes them unsuitable for complex patterns, e.g. functions which combine flat regions with high-variability ones. Recent advances have shown that any number of GP models can be stacked to implement deep hierarchies. These hierarchical models maintain the main advantages of GPs while learning more abstract and complex models. Deep Gaussian Processes (DGPs) were first introduced in [29] in 2013, their probabilistic DL modelling was very promising but the inference procedure complicated. In 2017, [30] introduced the doubly stochastic variational inference model for DGPs which, since then, became the current state of the art for DGP inference.

GPs, but not DGPs, have been used for different tasks in seismic problems, although none of them have been ever used before for automatic seismic-event classification. Specifically, GPs have been used for regression in seismic problems with promising results in this field. The authors of [31] proposed a generative model for seismic monitoring. This model can recover weak events from the raw signal. They used GPs over wavelet parameters to predict detailed waveform fluctuations based on historical events, while degrading smoothly to simple parametric envelopes in regions with no historical seismicity. The authors of [32] proposed a new approximation for large-scale GPs, specifically for GP latent variable models (GPLVM). They proposed to approximate the marginal likelihood of the full GP via a random Markov field in which local GPs are connected by pairwise potentials. This approximation allows to efficiently perform inference for spatial data and it was applied successfully to seismic location. The authors of [33] used GP regression for anomaly detection, more specifically, for fault detection in seismic data. Since the used GPs expected smooth functions, their results show that fault points can be detected when the smooth trend of layers is disrupted by faulting.

This paper represents, to the best of our knowledge, the first contribution on the use of GPs and DGPs for automatic seismic-event classification. An approach that is tested here on the seismic dataset recorded at the *Volcán de Fuego de Colima*, in Colima (Mexico). Due to the complex character of this classification problem, the current state-of-the-art methods are based on hierarchical deep models. We show here that GPs outperform all the shallow classifiers and that they are competitive to DNNs. The experiments also show that the 2-layer DGP model outperforms DNNs, in particular in classes hardly represented. Additional experiments indicate that GPs and DGPs can learn good models even when the database is small. When data is scarce, GPs are the best performing models. With more data, deeper models, like the 4-layer DGPs provide better results. The study on the prediction confidence of each model shows that GP-based methods obtained probabilities closer to 1 than DNNs.

The rest of the paper is organized as follows. In section 2 we provide a brief introduction to both GPs and DGPs. This introduction is expanded in appendix Appendix A where a complete theoretical and intuitive description of GPs and DGPs for multi-class classification problems is included. In section 3, we carry out an insightful and exhaustive experimental analysis whose goal is to compare GPs and DGPs to current state-of-the-art both shallow and deep classifiers on the database recorded at *Volcán de Fuego de Colima*, in Colima (Mexico) [20]. Conclusions are drawn in section 4.

2. Deep Gaussian Process classifier

In this section we provide a brief introduction to the use of GPs and DGPs for multiclass classification problems. An extended and more detailed introduction can be found in appendix Appendix A.

A multiclass classification problem with K classes consists of N labeled instances $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$ where $\mathbf{x}_n \in \mathbb{R}^D$ is the feature vector and $y_n \in \{1, \ldots, K\}$ the class label of the n-th instance. For each instance $\mathbf{x}_n$, its label y_n is modeled using K latent variables $\mathbf{f}_{n,:} = \{f_k(\mathbf{x}_n)\}_{k=1}^K$ through a specific likelihood $\mathrm{p}(\mathbf{y}|\mathbf{f}_{n,:})$. In this work we utilize the robust max likelihood, which prevents overfitting in GPs.

In a GP based formulation of a supervised problem we assume that the distribution of $\mathbf{f} = (f_1, \ldots, f_n)^{\mathrm{T}}$ given $\mathbf{X}$ is a multivariate normal, where we assume zero mean for simplicity and a kernel function $k(\cdot, \cdot)$ defines our covariance matrix. In this paper we use the squared exponential kernel (SE) defined as $k_{\mathrm{SE}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(\frac{-||\mathbf{x}_i - \mathbf{x}_j||^2}{2l^2}\right)$, where the parameters σ and l will be estimated from the observations. See Figures A.8, A.9 and A.11 in appendix Appendix A for a better understanding of GPs. For scalability, we also defined $M \ll N$ inducing points $\mathbf{u}_m$ which are the realization of the GP in the locations $\mathbf{z}_m$, that is, $\mathbf{u}_m = \mathbf{f}(\mathbf{z}_m)$. The inducing points summarize information from the entire dataset in a few points. Their locations are learned in the optimization process too. The posterior distribution for this model is not tractable so an approximate inference method has to be used. In this work, we follow the scalable variational inference for GPs

(SVGP) [34]. From here on, and to make explicit the inference procedure used, we will refer to the single-layer GP as SVGP.

SVGPs are flexible non-parametric probabilistic models very frequently used in classification and regression problems. However, these models can only represent a restricted class of functions. To overcome this limitation, hierarchical models based on GPs were proposed [29]. DGPs use the outputs of a standard SVGP as the input to another SVGP. If this is repeated L times, we obtain a hierarchy of SVGPs that is known as a DGP with $L + 1$ layers. Due to its hierarchical structure it achieves a greater level of abstraction and can capture more complex patterns. See Figures A.13 and A.14 in appendix Appendix A for a better understanding of how DGPs tackle the complexity in a toy example. Volcano seismic signals are very complex with classes that are difficult to distinguish. We will see that DGPs are very suitable for our classification task.

3. Practical Application: automatic events classification for the 'Volcán de Fuego de Colima', México

3.1. Database description

Section 3 analyses the performance of SVGP and DGP methods for classification of volcano-seismic events. Classification experiments are carried out using a database of 9.332 seismic events registered at the *Volcán de Fuego de Colima* in México [35]. These registers and their labels are the result of a careful and demanding process of expert analysis and review, to eliminate human artefacts and noise, to identify source mechanisms, and to analyse how site and path effects can influence waveforms. The labeled database contains 7 different events (classes) with diverse spectral and temporal characteristics, associated to 7 corresponding source mechanisms (REG, VTE, LPE, TRE, EXP, COL, and NOISE). They can be grouped as follows:

i. Events originated by fractures of solid materials in the earth's crust: *Regional Earthquakes* (REG) and *Volcano-tectónic Earthquakes* (VTE). As a result of the fracture, elastic waves containing P- and S-wave components associated respectively to longitudinal and shear displacements are generated. If the fracture occurs in the surrounding of the volcano, the event associated is identified as 'VTE', and contains high frequencies reaching up to 40 Hz with durations from a few to tens seconds. On the other hand, fractures that might occur in fault planes beyond the volcanic region, can be registered by the seismometers in the volcanic area, being labeled as 'REG'. REG events contain frequencies lower than those of VTEs, because the higher ones have been absorbed through the propagation path from the fracture source location to the registering station. The database used in this work contains 1.738 VTEs and 455 REGs.

ii. When no fracture occurs, volumetric modes of deformation of the volcanic structure (often triggered by displacements of water, gas, or mamga), produce *Long Period Events* (LPE). They show frequencies of a few Hz (between 1 and 6 Hz for the Volcan de Colima) and durations of a few seconds. Having spectral characteristics and source mechanisms similar to those of LPEs but much longer duration, *Volcanic Tremors* (TRE) identify a series of harmonic signals with sustained amplitude and variable duration from minutes to hours. The database used in this work contains 2.699 LPEs and 1.170 TREs.

iii. There are also certain events associated to the external activity of the volcano. Often, sudden emissions of gas and ash to the atmosphere occur, and are recorded by seismometers, receiving the name of *Explosions* (EXP). They are characterized by a short-duration LPE, followed by high-frequency signals with a narrow energy peak that can reach up to 20 Hz. Surface lava movements, or *Lava Flows*, (COL) with durations of minutes and frequencies between 5 and 10 Hz, are also associated to the external dynamics of the volcano. The database used in this work contains 2.699 LPEs, 278 EXPs, and 1.406 COLs.

iv. Finally, seismic noise (NOISE) registered by stations in absence of volcanic source mechanisms, presents diverse amplitudes, frequencies and durations depending on its nature (wind, sea, rain, cultural noise...). The database used contains 1.586 NOISE examples.

For comparison purposes, we follow the approach in [20]. The events used to feed the models are parametrized to create input feature vectors with 21 features. Seismic registers are first filtered in the band 1 to 25 Hz. Then, regardless of their duration, they are divided into 3 segments of equal length (beginning, central part and ending of the event). After that, following a common parametrization in the field, for each segment, a feature vector of 5 Linear Predictive Coding (LPC) coefficients is calculated. The 15-features vector so built is completed with 6 statistical features proposed in [36]. Features 16 to 18, parametrize the impulsiveness of the signal in the time domain by calculating the 20^{th}, 50^{th}, and 80^{th} cumulative-sum percentiles of the signal's amplitude. Following the same approximation in the frequency domain, features 19 to 21 calculate the 20^{th}, 50^{th}, and 80^{th} cumulative-sum percentiles of the signal's power spectral density.

3.2. Experiments description

The chosen methods for this study are the following: the single-layer SVGP (SVGP) and the 2-layer (DGP2), 3-layer (DGP3), and 4-layer (DGP4) DGPs. We also include an exhaustive and insightful comparison to state-of-the-art shallow and deep classifiers. The selected shallow classifiers are Support Vector Machine with linear (SVM-Lin) and radial (SVM-Rad) kernels, Random Forest (RF), and a single-layer MLP. The deep classifiers are the following deep neural networks (DNNs): Deep Belief Network with 2 (DBN-H2) and 3 (DBN-H3) hidden layers, and Stacked Denoising Autoencoder with 2 (sDA-H2) and 3 (sDA-H3) hidden layers. Configuration details for these classifiers, which were tuned performing grid searches for the optimal number of neurons per layer, are fully described in detail in [20].

The dataset is carefully split into four folds to perform a four-fold cross-validation analysis. Taking into account the unbalanced nature of the classes of volcano-seismic events, folds are carefully checked to ensure well-balanced statistically representative experiments. For the sake of comparison between GPs and other deep learning approaches, the exact same database and folds used in [20] are used in the present experiments. Given the need to tune the system architecture, on each round of the cross-validation two folds are used for training, one to search the optimal configuration (grid search for the possible numbers of neurons per layer) and another to evaluate the classification results.

We use three different metrics to assess the performance: the f1 score, accuracy, and log loss. We define the multiclass accuracy and log loss as,

$$\text{accuracy} = \frac{\text{No. events classified correctly}}{\text{Total no. events}} \tag{1}$$

$$\text{log loss} = \frac{-1}{N} \sum_{n=1}^{N} \log(\mathrm{p}(y_n) \cdot \mathbf{e}_{k_n}) \tag{2}$$

being N the total number of events, the dot $(\cdot)$ denotes the scalar product and $\mathbf{e}_{n_k}$ the one-hot encoding vector of the true class of the n-th instance. Notice that the accuracy is the percentage of global success and the log loss measures not only the success but the confidence of the classifier. We define the f1 score per class using the true positives (TP), false negatives (FN) and false positives (FP) as

$$\text{f1 score} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FN} + \text{FP}}, \tag{3}$$

and then, we take the average of them to obtain the multiclass macro average f1 score. Notice that this metric penalizes the misclassification of samples coming from underrepresented classes while the log loss and accuracy do not.

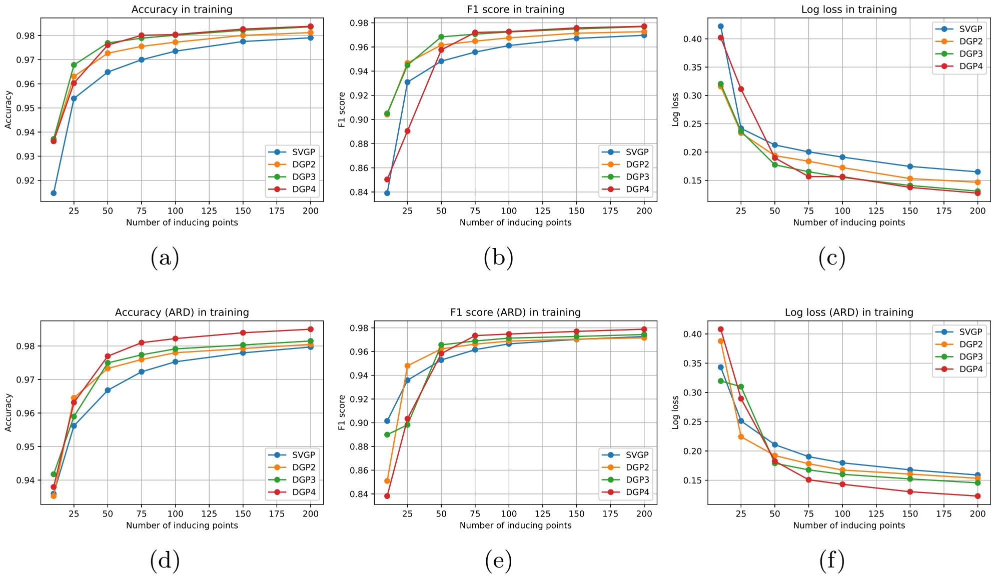
To provide a deep insight into the particular needs in classification of volcano-seismic events, the rest of the experimental section has been structured as follows. Firstly in subsection 3.3, we study the behavior and selection of hyperparameters using the validation set. In addition to the selection of the model configuration, this experiment also provides a better understanding of the presented models. Then, in subsections 3.4 and 3.5, we assess the generalization capability of the models on the test set. Finally, additional experiments of relevant interest in the area of knowledge are reported. Given the lack of large high-quality labeled databases, the robustness of the classification against different sizes of the dataset is studied in subsection 3.6. In addition, in order to handle the difficulties to classify some events that could correspond to diverse source mechanisms (including overlapped ones), confidence measures of the predictions for the different classifiers are studied in subsection 3.7.

3.3. Selection of SVGP and DGP hyperparameters

In contrast to other classifiers where an exhaustive grid search is used for hyperparameter tuning, in GP-based methods almost all the parameters are estimated automatically and learned through an optimization process. Following common practice [30], we utilize the same number of hidden units in each layer. Since in this problem we have a reduced number of features, we set it to 7 after an empirical search. We use the SE kernel defined in eq. (A.4). In this model the lengthscale l associated to all the features is the same. This is very useful to avoid overfitting in scenarios with small databases but features frequently have different discriminative power. In the experiments, for an exhaustive comparison, we also use the Automatic Relevance Determination (ARD) model, whose kernel is defined by:

$$k_{\text{SE-ARD}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\sum_{d=1}^{D} \frac{||x_i(d) - x_j(d)||^2}{2l_d^2} \right), \tag{4}$$

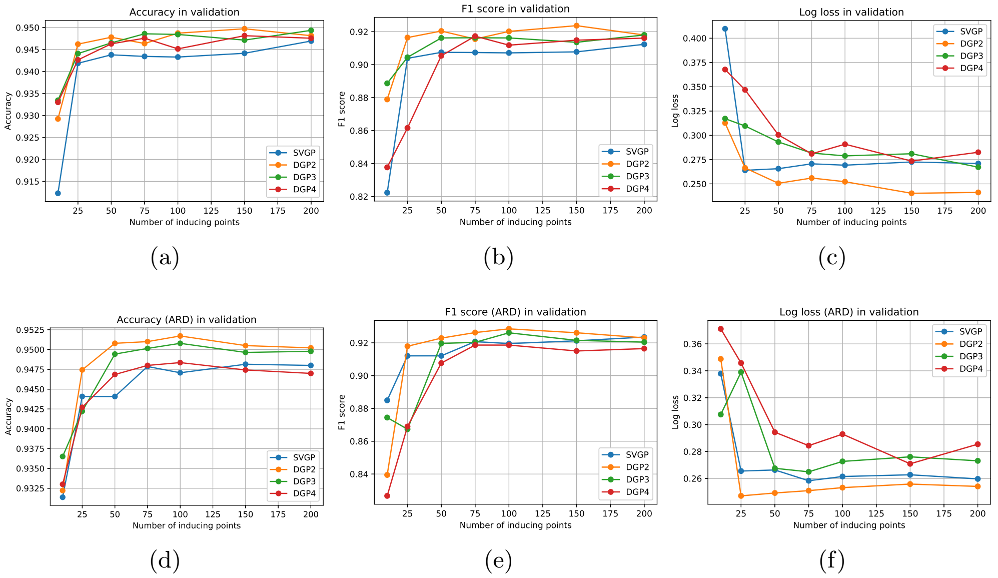
where $x_i(d)$ is the d-th coordinate of the feature vector $\mathbf{x}_i$ and l_d is its lengthscale parameter.



(a) (b) (c)

(d) (e) (f)

Figure 1: Results on the training set for GP-based models. Using an SE kernel (first row) and an SE-ARD one (second row). Each column corresponds to a different metric (from left to right): accuracy, f1 score and log loss.



(a) (b) (c)

(d) (e) (f)

Figure 2: Results on the validation set for GP-based models. Using an SE kernel (first row) and an SE-ARD one (second row). Each column corresponds to a different metric (from left to right): accuracy, f1 score and log loss.

To adjust the number of inducing points we choose the following grid analysis: 10, 25, 50, 75, 100, 150 and 200 points. We report the following metrics for every combination of inducing points and kernel (SE or SE-ARD): accuracy, f1-score and log loss, for both training and validation sets.

Results on the training set are shown in figure 1. As the number of layers and inducing points increases, the models perform better except for a slightly noisy behavior

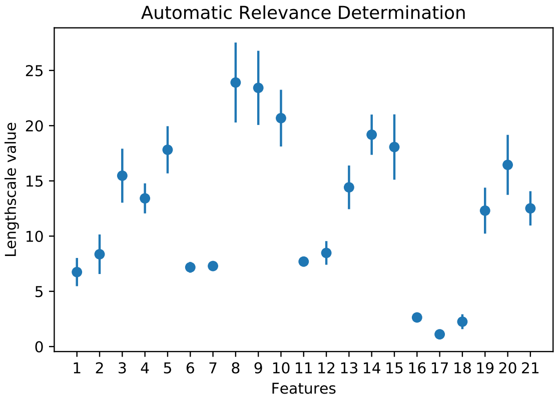


Figure 3: Estimated lengthscale values for the SVGP in test. The points represent the averaged values and the bars the standard deviation. Lower (downwards) represents more importance of that feature for the classifier.

Table 1: Averaged performance in test: f1 score per class, macro-average f1 score and accuracy.

	Noise	EXP	REG	COL	VTE	TRE	LPE	*Macro f1*	*Accuracy*
No. Events	1586	278	455	1406	1738	1170	2699		
SVM-Lin	0.9685	0.6639	0.9103	0.9419	0.9342	0.808	0.9275	0.8792±0.0045	0.9155±0.008
SVM-SE	0.9686	0.7186	0.8881	0.9475	0.9339	0.8463	0.9387	0.8917±0.0084	0.9232±0.0076
RF	0.9653	0.7188	0.9013	0.9478	0.941	0.8641	0.9413	0.8971±0.0093	0.928±0.0061
MLP	0.9711	0.7533	0.9003	0.9645	**0.9468**	0.8667	**0.9485**	0.9073±0.0068	0.9373±0.007
SVGP	**0.974**	**0.8002**	**0.9113**	**0.9756**	0.9451	**0.8927**	0.9419	**0.9201**±0.0078	**0.9408** ±0.0027
Rel. Impr. (%)	0.2986	6.2259	1.2218	1.1509	-0.1796	2.9999	-0.6958	1.4108	0.3734

when few inducing points are used. On the validation set, see figure 2, more complex architectures do not lead to better models. Once we have a sufficiently high number of inducing points, we cannot capture more information to have a better performance in the validation set. To summarize the essential information of the analysis, 100 points are enough.

We also compare both the SE and SE-ARD kernels to find out whether there are features more relevant than others. We can see that the ARD results are in general slightly better. For example, looking at figure 2, SVGP, DGP2, and DGP3 reach a 0.92 f1 score value with ARD, while for SE DGP2 hardly reaches this value. Such improvement will be helpful when detecting underrepresented classes since this global metric gives weight to them.

To conclude this section. In general, the SE-ARD kernel model performs better than SE. Furthermore, SE and SE-ARD become stable once they reach 100 inducing points. Based on these results, we chose the SE-ARD kernel and 100 inducing points as hyperparameters of the SVGP and DGP models for the subsequent evaluation of the test sets.

3.4. Performance of shallow classifiers

In this subsection, we assess the generalization capability of shallow SVGP with 100 inducing points and SE-ARD kernel against the shallow state-of-art classifiers: SVM with linear (SVM-Lin) and SE (SVM-SE) kernels, RF with 120 estimators and a single-layer MLP.

In Table 1, we report per class and global f1 scores, and accuracy. Additionally, we calculated the 95% confidence interval for the global metrics. The number of events per class and the relative improvement (Rel. Impr %) of the SVGP compared to MLP are also analyzed. The different classes of events present diverse difficulties for classification, depending on the number of instances per class and the variability and specificity of their associated features. EXP and REG are the most challenging types of events. There are two reasons for their lower classification results. First the number of examples per class is very small compared to the rest of classes. Second, their spectral and temporal properties are similar to those of other classes, making the discrimination more challenging. We can see this fact clearly reflected in the f1 score per class in Table 1.

SVM-Lin is the worst performing model although it is competitive compared to SVM-SE and RF. Furthermore, as it can be seen, SVGP outperforms every shallow method. In particular, although MLP is better for two classes, SVGP achieves the best accuracy, working specially well on the challenging classes, i.e. EXP and REG. This behavior is a consequence of using non-parametric models against those with a large number of parameters.

The usage of SE-ARD provides a better model convergence, avoiding certain noise introduced by less discriminative or redundant features. Besides, it points out which are these more noisy features and which are the most effective ones. We estimated the lengthscale for every dimension of the input feature vector. Lower values indicate higher discriminative power of these features. Figure 3 shows the lengthscale values for the 21 features per event used to feed the classifiers described in section 3.1. For each segment of event (beginning, central and final part), LPC coefficients 1 to 2 (features 1,2,6,7,11,12 in the feature vector) have the highest relevance. In particular, the shortest lengthscale values correspond to time domain (features 16 to 18), while LPC coefficients 3 to 5 in the central segment of signal (features 7 to 9) present the smallest discriminative relevance. Notice that the most discriminative features have shorter deviation, being relevant across different folds while less discriminative features do not.

Experiments in this subsection show that using a SVGP we are able to outperform widely used shallow methods such as SVM, RF or MLP, mainly when the number of events is scarce. Furthermore, information about the discrimintative potential of the input features can be extracted when using SVGP and the SE-ARD kernel.

3.5. Performance of deep classifiers

The complexity of seismic events motivates the use of DL although the reduced number of data may make them prone to overfitting. As we will see, the use of deep non-parametric models overcomes this problem. In this subsection, we compare the best shallow method, i.e. the SVGP, together with its hierarchical extensions, DGPs, i.e. DGP2, DGP3 and DGP4, to the DBN and the sDA reported in [20]. Both DBN and sDA with two and three hidden layers denoted by DBN-H2, DBN-H3, sDA-H2 and sDA-H3, respectively, are considered. These models use the log loss as the cost function minimizing it with stochastic gradient descent. To avoid overfitting, an early stopping criterion and dropout with $p = 0.20$ are used.

In Table 2, we report per class and global f1 scores and accuracy. Additionally, we calculated the 95% confidence interval for the global metrics. For the sake of comparison, the relative improvement of the best DGP technique over the best DNN technique is also presented for each class of events. The results confirm the advantages of using deep models for this problem. The reported metrics are better than those in the previous section except the ones related to the SVGP. SVGP is very competitive to DNNs. It has a lower accuracy (0.9408) than the best DNN, sDA-H2 (0.9432), but the global f1 score is

Table 2: Averaged performance in test: F1 score per class, macro-average F1 score and accuracy.

	Noise	EXP	REG	COL	VTE	TRE	LPE	*macro F1*	*Accuracy*
No. Events	1586	278	455	1406	1738	1170	2699		
DBN-H2	0.9756	0.7542	0.9143	0.9729	0.9430	0.8898	0.9485	0.9140±0.0065	0.9404±0.0068
sDA-H2	0.9741	0.7778	0.9151	0.9697	0.9484	0.8978	**0.9511**	0.9192±0.006	0.9432±0.0066
DBN-H3	0.97	0.77	0.89	0.97	**0.95**	0.89	0.95	0.91±0.0074	0.9387±0.0069
sDA-H3	0.97	0.78	0.91	0.97	0.94	0.89	0.95	0.92±0.0054	0.9410±0.0068
SVGP	0.974	0.8002	0.9113	0.9756	0.9451	0.8927	0.9419	0.9201±0.0078	0.9408±0.0027
DGP2	0.9789	**0.8391**	0.9175	0.9789	0.9461	**0.9103**	0.9478	**0.9312**±0.0034	0.9477±0.0043
DGP3	0.9765	0.8095	0.9031	0.9723	0.943	0.899	0.9447	0.9211±0.0066	0.9419±0.0058
DGP4	**0.982**	0.8264	**0.9182**	**0.9803**	0.9497	0.9055	0.9472	0.9299±0.0095	**0.9479**±0.0065
Rel. Impr. (%)	0.6560	7.5769	0.3388	0.7606	-0.0316	1.3923	-0.3470	1.2174	0.4983

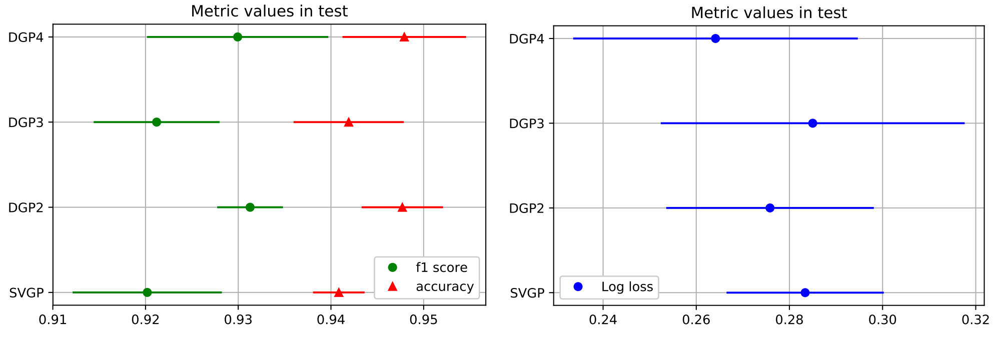
similar in both, i.e SVGP (0.9201) and sDA-H3 (0.92). Specifically, SVGP outperforms the DNNs for the class EXP showing the capacity of GP models to handle difficult and imbalanced datasets. This fact is confirmed looking at the best DGP models, DGP2 and DGP4. Both models outperform the rest in accuracy and f1 score obtaining the best global accuracy value and also performing better in difficult and less represented classes. In addition, DGP2 is statistically significant with respect to DNNs since their confidence intervals do not overlap. Regarding the f1 score per class, the best GP-based models, i.e. DGP2 and DGP4, perform remarkably well for difficult classes. Specially, they work notably well in EXP, COL, REG and TRE while DNNs only outperform DGPs in the LPE and VTE classes which, together with NOISE, are more easy to identify. It is also worth to point out that DGP2 is the best classifier identifying EXPs (0.8391), with a high relative improvement (7.57%). As it can be observed, the relative improvement is inversely related to the number of events in the class, and in these cases the difference seems significant. The improvement in the detection of EXP obtained when DGPs are used is very important in monitoring volcanic environments because together with the LPE and VT they are often precursors of volcanic activity [2].

In figure 4, we depict accuracy, f1 score, and log loss for the GP-based models. Average values of the four cross-validation experiments are depicted with a dot, within an interval line covering the results's standard deviation for the four experiments. This figure provides a better understanding of the results shown in Table 2. DGP2 and DGP4 are the best performing models while SVGP and DGP3 perform worse. In contrast to DGP2, DGP4 suffers from larger standard deviation values. This higher variance in the results indicates the presence of overfitting in complex models. In this sense, DGP2, with a very good performance too, appears to be the model with the greatest generalization capability.

In conclusion, deep GPs capture the complex patterns of seismic signals better than DNNs, benefiting from the use of full probabilistic non-parametric models. These results prove the adequacy of the GP-based models for classification of volcano-seismic events. Furthermore, GPs not only perform well globally but, specially, on these important classes. Finally, DGP2 is the best performing model with good global accuracy, a reduced variance and with the best result on the most challenging class, i.e. EXPs.

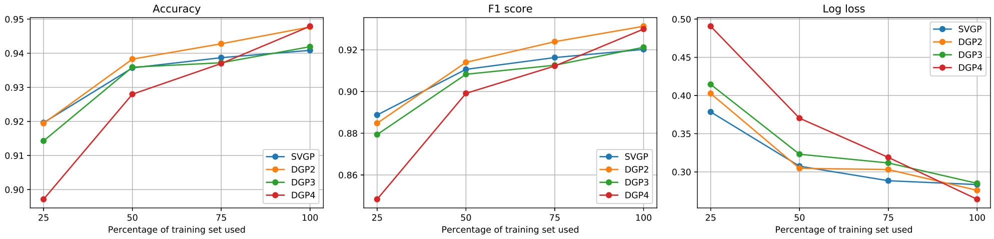
3.6. Robustness to the size of the training set

In the previous subsection, we showed the superiority of GP-based models against DNN ones in test performance. In seismic data, usually, we only have access to a small amount of labeled data, so it is also interesting to analyze the behavior of the studied methods when only a small dataset is provided. In this subsection, we vary the amount of



(a) (b)

Figure 4: The points represent average performance on the test set and the bars indicate standard deviation: (a) F1 score and accuracy; (b) log loss.



(a) (b) (c)

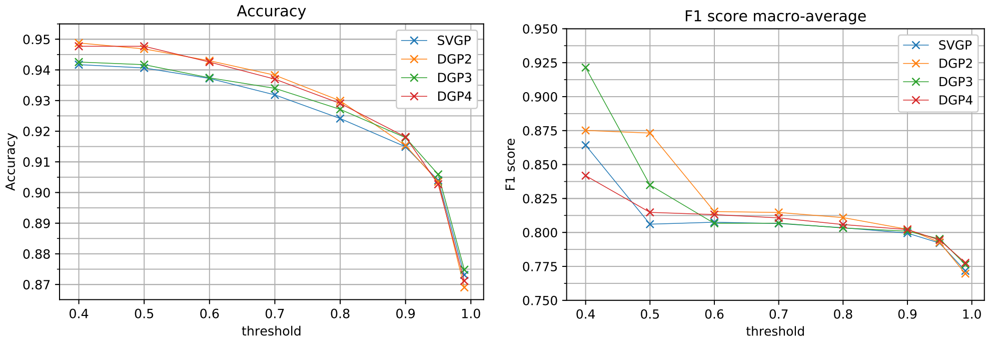
Figure 5: Accuracy, f1 score and log loss metrics varying the percentage of training samples available: 25%, 50%, 75%, and 100% of the training set.

training data available, using: 25%, 50%, and 75% of the whole dataset. The experiment reveals more about the adequacy of the proposed approach in scenarios where data are scarce. Figure 5 shows the accuracy, f1 score, and log loss performance of GP-based models. We can clearly see the need of data as the depth of the model increases. When only 25% of the training set is used, DGP4 performs poorly in contrast to the goodness of SVGP and DGP2. The SVGP performance does not improve much with the increase in data, in fact, it is the worst model with the entire dataset. In contrast, DGP4 improves enormously as the percentage of data increases. This fact suggests that shallow models are better in scenarios with small datasets while deeper models such as DGP4 play an interesting role when more data are available. We also find that DGP2 performs very well through the different experiments achieving very good results both with less and more data. Table 3 provides an accuracy comparison between the DNN values reported in [20] and those obtained by our GP-based models. Relative improvements of the best DGP model over the best DNN model are also described. The superiority of GPs is clear. For 25% and 50% of the data, DGP4 has not yet learned a good model, being inferior to the best DNN. However for 25%, 50%, and 75% all GP models outperform the best DNN.

As DNNs tend to overfit due to the huge amount of trainable parameters, they are more sensitive to smaller database sizes. In contrast, GPs use prior knowledge which acts like a strong regularization. They learn a good model even when a reduced dataset is provided. In summary, as the experiment confirms, GPs perform very well, and better

Table 3: Accuracy metric varying the percentage of training samples available among the 25%, 50% and 75% of the training set. The 100% corresponds to the entire training set.

Accuracy	25%	50%	75%	100%
DBN-H2	0.9069	0.9306	0.9283	0.9404
sDA-H2	0.9017	0.9121	0.9283	0.9432
DBN-H3	0.9125	0.922	0.928	0.9387
sDA-H3	0.9077	0.9209	0.9297	0.941
SVGP	**0.9196**	0.9356	0.9386	0.9408
DGP2	0.9194	**0.9382**	**0.9427**	0.9477
DGP3	0.9142	0.9359	0.9372	0.9419
DGP4	0.8971	0.9279	0.9369	**0.9479**
Rel. Impr. (%)	0.7562	0.8167	1.3983	0.4983



(a) (b)

Figure 6: Accuracy and f1 score metrics varying the classification probability threshold. A sample is predicted if the output probability of the highest class probability is higher than the selected threshold, otherwise this sample is unclassified.

than DNNs, for all data sizes.

3.7. Evaluating the confidence in the predictions

In volcano-seismic applications, it is of paramount importance to analyze the confidence of class predictions. Classification results are often used in early-warning tasks: detecting sequences of certain events which are precursors of eruptions; so trustable predictions together with a good quantification of their uncertainty are of high interest to design early warning systems. In this section, to analyse the quality of the predictions, we introduce a decision threshold over the probabilities output of the classification systems. By considering as classified only events assigned to a class with probability greater than the threshold and varying this threshold we can increase the confidence of the system. Two studies are performed:

i) First, we study accuracy and f1 score when different threshold values are used (0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0,95 y 0.99). The results are shown in figure 6. As we increase the threshold we are predicting less examples correctly, therefore accuracy and f1 score decrease. We can see that most samples are predicted with at least a 0.99 probability, we misclassify only the 7% if we change the threshold from 0.4 to 0.99. Note that 0.99 is a very demanding threshold being most samples predicted with very high probability (close to 1). The faster decrease of f1 score suggests that there is more uncertainty in the less represented classes.

In accuracy, for low thresholds, i.e. 0.4, 0.5, and 0.6, DGP2 and DGP4 classify more events correctly than SVGP and DGP3 but with higher thresholds this difference decreases (or is reduced). Regarding f1 score, we observe the same behavior. So the number of events classified with high probabilities is almost the same in all cases, but DGP2 and DGP4 are able to give more confidence to doubtful samples.

ii) For a complete understanding of the classifier confidence, figure 7 shows the distribution of the predicted probabilities per class. X-axis represents the probability of belonging to the class predicted. Y-Axis represents the cumulative density function of these probabilities for each class of events. Comparing the probability CDFs for different models, the figure provides information about how trustable the different classifications are.

Firstly, we confirm that few samples are predicted with very low probability; indeed, the most of the predictions are close to 1. This fact confirms that the models are confident on the predictions. They define good decision boundaries and identify every class well. All classifiers have a similar performance except for EXP and TRE; as we saw in section 3.5, both are the most difficult types of events. For these events, DGP2 and DGP4 perform better than SVGP and DGP3. This fact matches with the log loss reported in figure 4.

In figure 4 of paper [20], the authors reported the same cumulative density functions of classification probabilities values, for DNNs. We can see that GP-based methods outperform DNN ones in this experiment. For example, in EXP, the difference is quite clear. 50% is predicted with a 0.9 probability or more by the DNNs, in contrast, GPs predicted more than 60% with high probability, i.e. 0.9 or more.

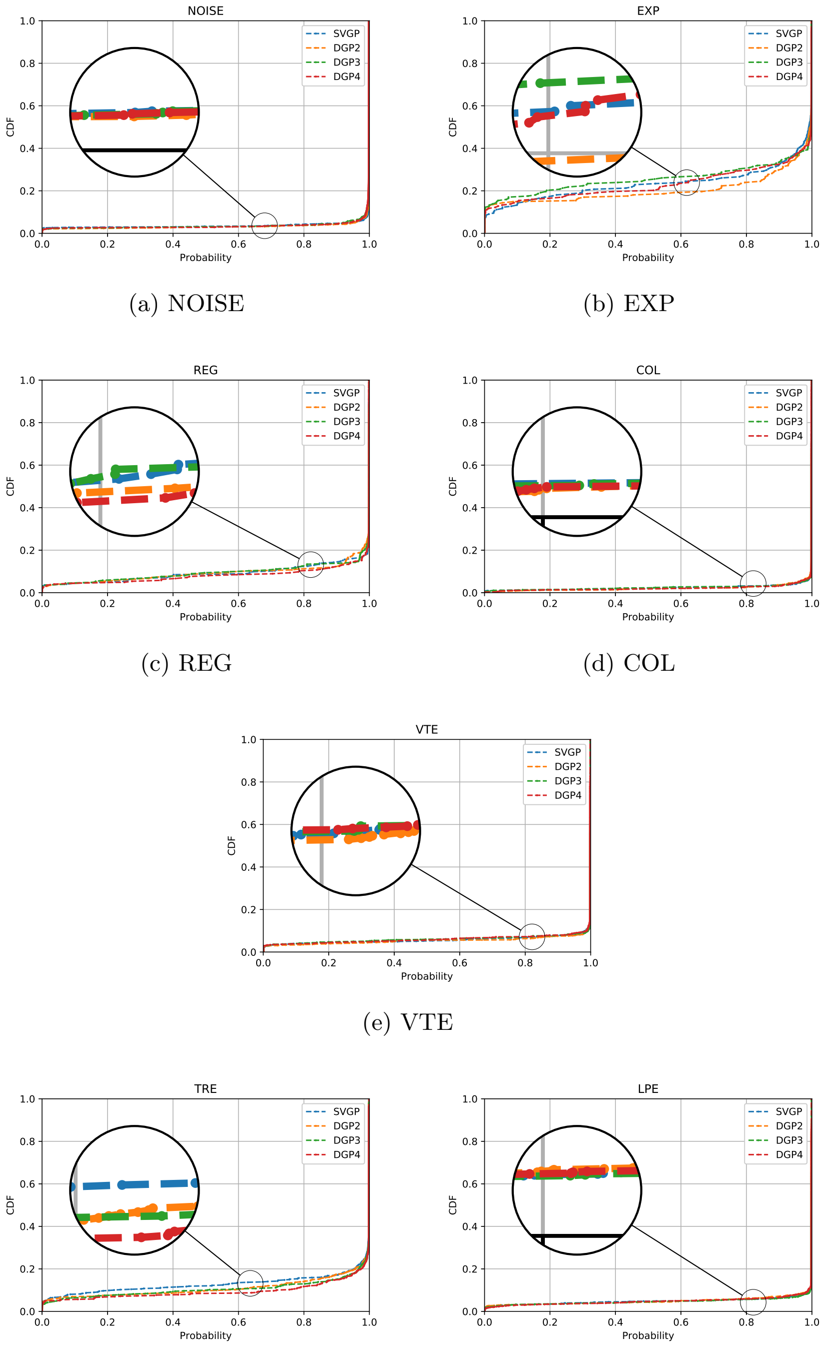
In conclusion, in this work, we observed that, for this database, the probabilities given by the GPs are more trustworthy than the ones provided by the DNNs and the best performing GP-based model, i.e. DGP2, is also the most confident.

4. Conclusion

In this work, we have introduced to the seismic community the usage of SVGPs and, their hierarchical extension, DGPs for automatic volcano-seismic event classification. We tested them on the seismic database recorded at *Volcán de Fuego de Colima*, in Colima (Mexico).

Due to the complexity of this problem, state-of-the-art methods are based on hierarchical deep models, i.e. DNNs. However, they require more data than usually available. The obtained results indicate that SVGPs outperform all the shallow classifiers. Moreover, they are competitive to DNNs. The 2-layer DGP outperforms DNNs avoiding overfitting. It attains both good accuracy and f1 score, and performs better than DNNs on difficult classes.

We have proven the adequacy of GPs with additional experiments. The experiments indicate that they can still learn good models even when the database is small. When data is scarce SVGP was the best performing method. Besides, with more data, deeper models, like 4-layer DGPs are an interesting option with promising results. In general, the 2-layer DGP performed very well through different percentages of training data.



(a) NOISE

(b) EXP

(c) REG

(d) COL

(e) VTE

(f) TRE

(g) LPE

Figure 7: Cumulative distribution function of the probabilities given by the GP classifiers per class. In the Y-axis we represent the proportion of samples of that class with a certain predicted probability or less. In the X-axis we represent these predicted probabilities.

Finally, we carried out an exhaustive study on the prediction confidence. GP-based methods obtained probabilities closer to 1 than DNNs.

These experiments suggest that GP-based methods are able to classify very well seismic events, specially interesting classes like EXP, REG and LPE, even when data is scarce. Besides, they take into account the model uncertainty, being a trustworthy system for volcanologists. In short, we have shown that GPs and DGPs can be applied with success to seismic problems.

Appendix A. Detailed introduction of Gaussian Processes and Deep Gaussian Processes

In this appendix, we provide a more detailed introduction to the use of GPs and DGPs for multiclass classification problems. We explain their probabilistic formulation, provide some intuition and examples of them, and describe how inference is carried out. An in-depth study of the inference methods followed here can be found in [34] for GPs and in [30] for DGPs.

A multiclass classification problem with K classes consists of N labeled instances $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$ where $\mathbf{x}_n \in \mathbb{R}^D$ is the feature vector and $y_n \in \{1, \dots, K\}$ the class label of the n-th instance. We define the $N \times D$ matrix $\mathbf{X}$ as the feature matrix where in the n-th row we have the feature vector of the n-th instance. In this work the features ($D = 21$) are extracted from the raw signal, more information about them is provided in subsection 3.1. We also define $\mathbf{y}$ the vector that gathers the labels of the samples. Once the supervised classifier is trained, it is able to provide the class label y_* for any unseen instance $\mathbf{x}_*$.

Appendix A.1. Single-layer GPs

For each instance $\mathbf{x}_n$, its label y_n is modeled using K latent variables $\mathbf{f}_{n,:} = \{f_k(\mathbf{x}_n)\}_{k=1}^K$ through a specific likelihood $\mathrm{p}(\mathbf{y}|\mathbf{f}_{n,:})$. The likelihood squashes the values of the latent variable defined in $\mathbb{R}$ to the $[0, 1]$ interval. Notice that this likelihood plays a similar role as the output neurons play in DNNs. For example, the so extended softmax function can be used here. In this work we utilize the robust max likelihood, which prevents overfitting in GPs. It is defined by

$$\mathrm{p}(y_n = k|\mathbf{f}_{n,:}) = \begin{cases} 1-\varepsilon & k = \underset{1\leq j\leq K}{\arg\max}\ \ \mathbf{f}_{n,j} \\ \frac{\varepsilon}{K-1} & \text{otherwise} \end{cases} \tag{A.1}$$

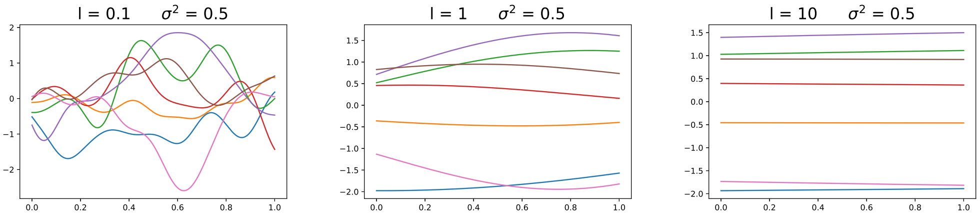
with $k \in \{1, \dots, K\}$ and $1 - \frac{1}{K} > \varepsilon > 0$ which is usually fixed to an small value, in this work it was fixed to 10^{-3}. For simplicity, we denote the latent variables by $f_k(\mathbf{x}_n) = f_{n,k}$.

We factorize the likelihood assuming that the class labels are independent for the different samples:

$$\mathrm{p}(\mathbf{y}|\mathbf{F}) = \prod_{n=1}^N \mathrm{p}(y_n|\mathbf{f}_{n,:}), \tag{A.2}$$

where $\mathrm{p}(y_n|\mathbf{f}_{n,:})$ is given by eq. (A.1). The $N \times K$ matrix $\mathbf{F}$ gathers the K latent variables for the N instances. The (n, k) term corresponds to the k-th latent variable for the n-th instance. The n-th row of $\mathbf{F}$ is denoted by $\mathbf{f}_{n,:}$, and the k-th column by $\mathbf{f}_k$.

Having defined the observation model, we now turn our attention to the definition of the prior model on $\mathbf{F}$. Notice that, at observation level, if the class of the $n-th$



(a) (b) (c)

Figure A.8: One dimensional example of a GP. We draw several samples from a GP with an SE kernel varying the lengthscale. Shorter values of the lengthscale l produce wriggly curves while larger values produce flat functions.

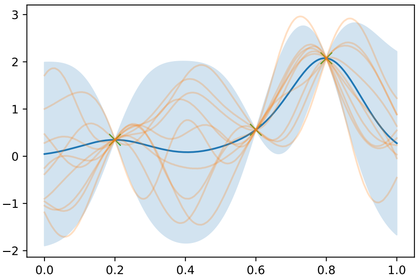


Figure A.9: One dimensional example of a GP with an SE kernel ($l = 0.1$ and $\sigma^2 = 1$). We have observed the values in $x_1 = 0.2$, $x_2 = 0.6$ and $x_3 = 0.8$. Then, we predict in 100 unobserved points X_* of the [0,1] interval given these observations. We draw $\mathrm{p}(F_*|X_*, X, F, \mathbf{\Theta})$ with $X = \{x_1, x_2, x_3\}$ and $F = \{f(x_1), f(x_2), f(x_3)\}$: the blue line is the mean and the blue shadow the 0.95 confidence interval. We also draw several samples from this distribution in orange. Observe that almost all samples are contained in the confidence interval.

sample is k, $f_{n,k}(\mathbf{x}_n)$ is larger than $f_{n,j}(\mathbf{x}_n), j \neq k$ and that we are assuming that a priori $\mathbf{f}_k$ and $\mathbf{f}_j, j \neq k$ are independent. So we need to model now the a priori behavior of each $\mathbf{f}_i, i = 1, \ldots, K$. We use a GP to define an a priori independent distribution for each column component of the latent matrix. A GP is an infinite collection of random variables in which every finite subset is Gaussian distributed. It can be seen as a prior over functions. So we assume that the columns of the latent variable $\mathbf{F}$, $\{\mathbf{f}_k\}_{k=1}^K$, follow independent GP priors. For every k, it imposes that $\{f_{n,k}\}_{n=1}^N$ follow jointly a Gaussian distribution $\mathcal{N}(\mathbf{f}_k|\mathbf{0}, \mathbf{K}_{\mathbf{XX}})$, where the covariance matrix is obtained using a kernel function $k(\cdot,\cdot)$ [24]. We can write the prior distribution of the latent function as

$$\mathrm{p}(\mathbf{F}|\mathbf{\Theta}, \mathbf{X}) = \prod_{k=1}^{K} \mathrm{p}(\mathbf{f}_k|\mathbf{\Theta}, \mathbf{X}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{f}_k|\mathbf{0}, \mathbf{K}_{\mathbf{XX}}), \tag{A.3}$$

where $\mathbf{\Theta}$ are the kernel hyperparameters. The covariance matrix $\mathbf{K}_{\mathbf{XX}} = \mathbf{K}(\mathbf{X}, \mathbf{X}) = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j}$ encodes the properties of the desirable function (e.g. smoothness).

In this work we use the squared exponential (SE) kernel:

$$k_{\mathrm{SE}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(\frac{-||\mathbf{x}_i - \mathbf{x}_j||^2}{2l^2}\right), \tag{A.4}$$

this kernel has a great power of representation and it is used in many different scenarios [24]. In this case, we have to estimate the lengthscale l and variance σ^2 hyperparameters.

Note that functions drawn from a GP with an SE kernel are infinitely differentiable leading to smooth functions which are desirable in most problems. In figure A.8 assuming that $x_1, \dots, x_N$ are 100 points evenly distributed in the interval $[0,1]$, we show several samples of a GP with different elections of the lengthscale, we can notice that it controls the level of smoothness. Larger values of this parameter produce flat functions while shorter values lead to wriggly functions. Assuming that the GP has been observed only at $x_1 = 0.2$, $x_2 = 0.6$, $x_3 = 0.8$ we show in figure A.9 the observed values together with the predicted values $f(X_*)$ for X_* being 100 points evenly distributed in the $[0,1]$ interval. The use of a GP imposes that $f(X_*), f(x_1,), \dots, f(x_3)$ are jointly Gaussian from which we can obtain the distribution of $f(X_*)$ given $f(x_1,), \dots, f(x_3)$. We also include their 0.95 confidence intervals.

The joint distribution of the probabilistic framework defined here is given by

$$\mathrm{p}(\mathbf{y}, \mathbf{F}, \mathbf{X}|\boldsymbol{\Theta}) = \underbrace{\mathrm{p}(\mathbf{y}|\mathbf{F})}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{F}|\mathbf{X}, \boldsymbol{\Theta})}_{\text{GP prior}}. \tag{A.5}$$

Approximate inference methods, such as Laplace Method or Expectation Propagation, have a computational cost of $\mathcal{O}(KN^3)$ because they involve the inversion of an $N \times N$ dimensional matrix. To amend this problem, we use the sparse approximation of GPs [34]. We define $M \ll N$ inducing points for each GP. These inducing points are latent variables, they are the values of the GP realization at the inducing point locations $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_M\} \subset \mathbb{R}^D$. We gather them in the $M \times K$ matrix $\mathbf{U}$. The (m, k) term corresponds to the k-th latent variable of the m-th inducing point. The m-th row of $\mathbf{U}$ is denoted by $\mathbf{u}_{m,:}$, and the k-th column by $\mathbf{u}_k$. As we have indicated these inducing points can be seen as $\mathbf{U} = \mathbf{F}(\mathbf{Z})$. We are summarizing the value of the true latent function through the inducing points so it is important to optimize on their location. It is expected that these optimal locations will end up close to informative places as the decision boundaries. The probabilistic model of the sparse approach is given by

$$\mathrm{p}(\mathbf{y}, \mathbf{F}, \mathbf{U}|\boldsymbol{\Theta}) = \underbrace{\mathrm{p}(\mathbf{y}|\mathbf{F})}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{F}|\mathbf{U}, \boldsymbol{\Theta})\mathrm{p}(\mathbf{U}|\boldsymbol{\Theta})}_{\text{GP prior}}, \tag{A.6}$$

Notice that

$$\mathrm{p}(\mathbf{y}, \mathbf{F}|\boldsymbol{\Theta}) = \int \mathrm{p}(\mathbf{y}|\mathbf{F})\mathrm{p}(\mathbf{F}|\mathbf{U}, \boldsymbol{\Theta})\mathrm{p}(\mathbf{U}|\boldsymbol{\Theta})\mathrm{d}\mathbf{U}, \tag{A.7}$$

and so the above factorization does not modify the modelling. Fortunately, it provides us with a tool to perform tractable inference. We show the probabilistic graphical model using inducing points in figure A.10.

In this work, we follow the scalable variational inference for GPs (SVGP) [34]. It will allow to estimate the model parameters $\boldsymbol{\Theta}$ and also to approximate the posterior distribution $\mathrm{p}(\mathbf{F}, \mathbf{U}|\mathbf{y}, \boldsymbol{\Theta})$ by the distribution $\mathrm{q}(\mathbf{F}, \mathbf{U})$. Using the joint distribution and Jensen's inequality we obtain the well known evidence lower bound (ELBO):

$$\log \mathrm{p}(\mathbf{y}|\boldsymbol{\Theta}) \geq \int \mathrm{q}(\mathbf{F}, \mathbf{U}) \log \frac{\mathrm{p}(\mathbf{y}, \mathbf{F}, \mathbf{U}|\boldsymbol{\Theta})}{\mathrm{q}(\mathbf{F}, \mathbf{U})} \mathrm{d}\mathbf{U}\mathrm{d}\mathbf{F}, \tag{A.8}$$

notice that this bound is valid for every $\mathrm{q}(\mathbf{F}, \mathbf{U})$ distribution. It is straightforward to see that maximizing the ELBO is equivalent to minimize the Kullback-Leibler divergence

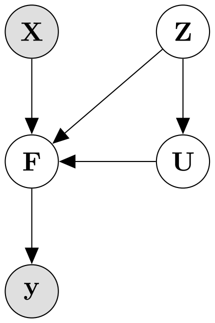


Figure A.10: Probabilistic graphical model of a scalable variational gaussian process (SVGP). Dark circles stand for observed variables while light circles stand for latent variables.

between $\mathrm{q}(\mathbf{F},\mathbf{U})$ and $\mathrm{p}(\mathbf{F},\mathbf{U}|\mathbf{y},\boldsymbol{\Theta})$. The SVGP approximation utilizes the following parametric form for q:

$$\mathrm{q}(\mathbf{F},\mathbf{U}) = \mathrm{q}(\mathbf{F}|\mathbf{U},\Theta)\mathrm{q}(\mathbf{U}) \tag{A.9}$$

$$\mathrm{q}(\mathbf{F}|\mathbf{U},\boldsymbol{\Theta}) = \mathrm{p}(\mathbf{F}|\mathbf{U},\boldsymbol{\Theta}) \tag{A.10}$$

$$\mathrm{q}(\mathbf{U}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{u}_k|\mathbf{m}_k,\mathbf{S}_k) \tag{A.11}$$

and then the ELBO can be rewritten as:

$$\begin{aligned}
&\log \mathrm{p}(\mathbf{y}|\boldsymbol{\Theta}) \geq \\
&\int \mathrm{q}(\mathbf{U})\mathrm{p}(\mathbf{F}|\mathbf{U}) \log \frac{\mathrm{p}(\mathbf{y}|\mathbf{F})\mathrm{p}(\mathbf{F}|\mathbf{U})\mathrm{p}(\mathbf{U})}{\mathrm{p}(\mathbf{F}|\mathbf{U})\mathrm{q}(\mathbf{U})} \mathrm{d}\mathbf{U}\mathrm{d}\mathbf{F} \\
&= \mathbb{E}_{\mathrm{p}(\mathbf{F}|\mathbf{U})\mathrm{q}(\mathbf{U})} \log \mathrm{p}(\mathbf{Y}|\mathbf{F}) + \mathbb{E}_{\mathrm{q}(\mathbf{U})} \left(\frac{\mathrm{p}(\mathbf{U})}{\mathrm{q}(\mathbf{U})} \right) \\
&= \sum_{n=1}^{N} \mathbb{E}_{\mathrm{q}(\mathbf{f}_{n,:})} \log \mathrm{p}(\mathbf{y}_i|\mathbf{f}_{n,:}) - \sum_{k=1}^{K} KL(\mathrm{q}(\mathbf{u}_k)||\mathrm{p}(\mathbf{u}_k)),
\end{aligned} \tag{A.12}$$

where KL is the Kullback-Leibler divergence. The derivation in eq. (A.12) allows to see the ELBO as the sum of two terms: the first one is a fidelity term imposing that the latent classifier must classify well, and the second one a regularization term over the latent variable in the inducing points. Our final goal then becomes to find the optimal kernel hyperparameters $\tilde{\boldsymbol{\Theta}}$, inducing locations $\tilde{\mathbf{Z}}$ and variational parameters of $\mathrm{q}(\mathbf{U})$, i.e. $\tilde{\mathbf{m}}_k, \tilde{\mathbf{S}}_k$, by maximizing the ELBO in eq. (A.12). Furthermore, since the ELBO factorizes over the instances, we can use mini-batches for optimizing this function reducing the computational cost, in this case considering that $M < N_b$, the computational cost is $\mathcal{O}(N_b M^2 K)$, where N_b is the mini-batch size.

Once the ELBO is optimized and the variational parameters computed, we can make predictions on an unseen test sample $\mathbf{x}_*$. The value of the latent variable $\mathbf{f}_*$ on this point $\mathbf{x}_*$ is given by

$$\begin{aligned}
\mathrm{p}(f_{*,k}|\mathbf{x}_*,\tilde{\boldsymbol{\Theta}},\mathbf{X},\mathbf{y}) &= \int \mathrm{p}(f_{*,k}|\mathbf{u}_k)\mathrm{p}(\mathbf{u}_k|\tilde{\boldsymbol{\Theta}})d\mathbf{u}_k \\
&\approx \mathbb{E}_{\mathrm{q}(\mathbf{u}_k)}\mathrm{p}(f_{*,k}|\mathbf{u}_k) \\
&= \mathcal{N}(f_{*,k}|\tilde{\boldsymbol{\mu}},\tilde{\boldsymbol{\Sigma}}),
\end{aligned} \tag{A.13}$$

where the mean and the covariance matrix are defined by:

$$\tilde{\boldsymbol{\mu}} = \mathbf{K}_{\mathbf{x}_*\tilde{\mathbf{Z}}}\mathbf{K}^{-1}_{\tilde{\mathbf{Z}},\tilde{\mathbf{Z}}}\tilde{\mathbf{m}}_k, \tag{A.14}$$

$$\tilde{\boldsymbol{\Sigma}} = k_{\mathbf{x}_*\mathbf{x}_*} + \mathbf{K}_{\mathbf{x}_*\tilde{\mathbf{Z}}}\mathbf{K}^{-1}_{\tilde{\mathbf{Z}},\tilde{\mathbf{Z}}}(\tilde{\mathbf{S}}_k - \mathbf{K}_{\tilde{\mathbf{Z}}\tilde{\mathbf{Z}}})\mathbf{K}^{-1}_{\tilde{\mathbf{Z}},\tilde{\mathbf{Z}}}\mathbf{K}_{\tilde{\mathbf{Z}}\mathbf{x}_*}). \tag{A.15}$$

Finally, the class label is obtained using

$$\mathrm{p}(\mathbf{y}_*) = \int \mathrm{p}(\mathbf{y}_*|\mathbf{f}_*)\mathrm{p}(\mathbf{f}_*|\mathbf{x}_*, \tilde{\boldsymbol{\Theta}}, \mathbf{X}, \mathbf{y})d\mathbf{f}_*, \tag{A.16}$$

this integral is intractable and it can be computed using numerical algorithms, e.g. Gaussian-Hermite quadrature.

In figure A.11, we depict a 1-D binary toy example to provide a better understanding of the SVGP model. The observation model is

$$\mathrm{p}(y_n|f(x_n)) = \left(\frac{1}{1+e^{-f(x_n)}}\right)^{y_n}\left(1 - \frac{1}{1+e^{-f(x_n)}}\right)^{1-y_n} \tag{A.17}$$

with $y_n \in \{0, 1\}, x_n \in \mathbb{R}$. Notice that here we only have one SVGP. The blue dots are class 0 and 1 observations, they have been observed at $x \in [0, 1]$. In figure A.11a, the blue line represents the mean of the posterior latent function distribution and the blue shadow the confidence interval. The wider this shadow the more uncertainty. We can see that there is more uncertainty in the middle of the interval because there are no observations there. This latent function takes values in $\mathbb{R}$ so it has to be squashed into the $[0, 1]$ interval using the likelihood. In figure A.11b, the black line goes from 0 to 1 and corresponds to the value of $\mathrm{p}(\mathbf{y}_*)$ for $y_* = 1$ in eq. (A.16). Notice how this value takes into account all the possible values of f_*.

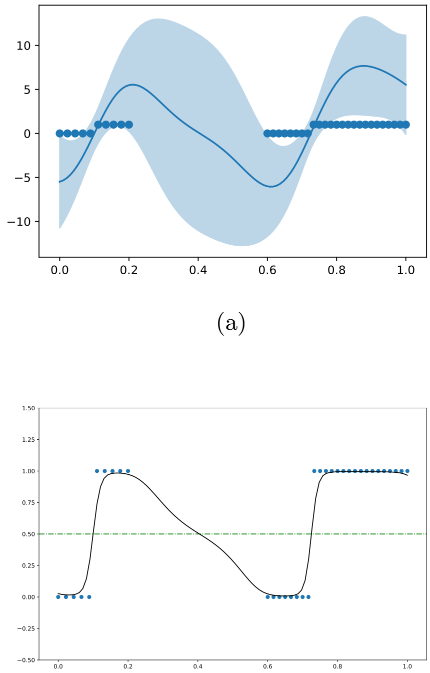
Appendix A.2. Deep Gaussian Processes

In this subsection, we detail the hierarchical extension of SVGP. Roughly speaking, the idea behind DGPs is to stack several SVGPs. If we use the output of one SVGP as the input of another SVGP and we repeat this procedure L times we define the $(L + 1)$ layer. DGP were first introduced in [29].

As it happens to SVGP, exact inference is also intractable for DGPs. In this work, we follow the doubly stochastic inference proposed in [30]. We introduce, at each layer l, M inducing points $\mathbf{U}^l$ at inducing locations $\mathbf{Z}^{l-1}$. The joint distribution of the probabilistic framework defined here is given by

$$\begin{aligned}\mathrm{p}(\mathbf{y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = &\underbrace{\prod_{n=1}^N \mathrm{p}(y_n|f_n^L)}_{\text{likelihood}} \\ &\times \underbrace{\prod_{l=1}^L \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1})}_{\text{DGP prior}},\end{aligned} \tag{A.18}$$

we consider $\mathbf{F}^0 = \mathbf{X}$, and each factor in the product is the joint distribution over $(\mathbf{F}^l, \mathbf{U}^l)$ of a SVGP in the inputs $(\mathbf{F}^{l-1}, \mathbf{Z}^{l-1})$, but rewritten with the conditional probability given $\mathbf{U}^l$. We introduce here the semicolon notation to clarify which are the inputs in



(a)

(b)

Figure A.11: One dimensional binary classification problem. The blue points represent the observations. In (a) we draw p(f_*): the blue line is the mean and the blue shadow the 0.95 confidence interval on the predictions. The classifier has more uncertainty in the region where there are no observations. In (b) we squash the latent function to the [0,1] interval, the black line is p($y_* = 1$).

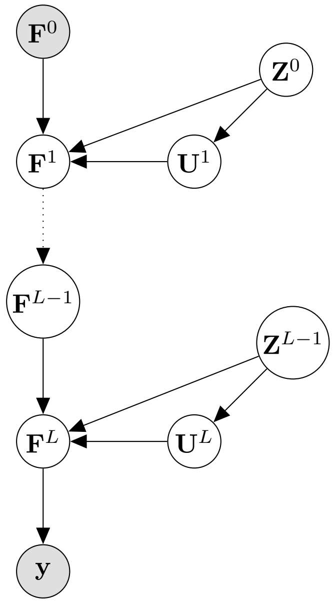


Figure A.12: Probabilistic graphical model of a deep gaussian process with L layers. Dark circles stand for observed variables while light circles stand for latent variables. The dotted arrow refers to the inductive process for building the general deep model.

the equations. We also consider the same amount of inducing points in every layer but notice that the hidden size of each layer can be different. $\mathbf{F}^l$ and $\mathbf{U}^l$ are $N \times D^l$ and $M \times D^l$ matrices, respectively. In this case, $\mathbf{Z}^{l-1}$ is a $M \times D^{l-1}$ matrix. We show the graphical probabilistic model in figure A.12, it illustrates the hierarchical construction of this architecture.

Following the same approach used in the single-layer case, we use variational inference to find a posterior distribution approximation $\mathrm{q}(\{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L)$:

$$\mathrm{q}(\{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = \prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{q}(\mathbf{U}^l). \tag{A.19}$$

where we impose the factorization $\mathrm{q}(\mathbf{U}^l) = \mathcal{N}(\mathbf{U}^l|\mathbf{m}^l, \mathbf{S}^l)$. The ELBO can then be written

$$\begin{aligned}\log \mathrm{p}(\mathbf{y}) \geq & \int \prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{q}(\mathbf{U}^l) \\ & \times \log \frac{\prod_{n=1}^{N} \mathrm{p}(y_n|\mathbf{f}_{n,:}^L) \prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1})}{\prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{q}(\mathbf{U}^l)} \\ & \times \prod_{l=1}^{L} \mathrm{d}\mathbf{U}^l \mathrm{d}\mathbf{F}^l \\ & = \sum_{n=1}^{N} \mathbb{E}_{\mathrm{q}(\mathbf{f}_{n,:}^L)}[\log \mathrm{p}(y_n|\mathbf{f}_{n,:}^L)] - \sum_{l=1}^{L} \mathrm{KL}(\mathrm{q}(\mathbf{U}^l)||\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1})).\end{aligned} \tag{A.20}$$

Now, we also estimate the model parameters for every layer, the variational parameters of $\mathrm{q}(\mathbf{U}^l)$ and the inducing point locations $\mathbf{Z}^{l-1}$. Again, the first term corresponds to a fidelity term and the second one to a regularization of the latent variable at each layer. In this case, the second term is tractable since it is the KL divergence between Gaussians. However, the first term involves the marginals of the posterior at the last layer, $\mathrm{q}(\mathbf{f}^L_{n,:})$ which is analytically intractable. Fortunately, it can be sampled efficiently using univariate Gaussians.

Marginalizing out the inducing points in eq. (A.19), the posterior distribution for the GP layers $\{\mathbf{F}^l\}_{l=1}^L$ becomes

$$\mathrm{q}(\{\mathbf{F}^l\}_{l=1}^L)=\prod_{l=1}^L \mathrm{q}(\mathbf{F}^l|\mathbf{m}^l,\mathbf{S}^l;\mathbf{F}^{l-1},\mathbf{Z}^{l-1})=\prod_{l=1}^L \mathcal{N}(\mathbf{F}^l|\tilde{\boldsymbol{\mu}}^l,\tilde{\boldsymbol{\Sigma}}^l), \tag{A.21}$$

where $[\tilde{\boldsymbol{\mu}}^l]_n = \mu_{\mathbf{m}^l,\mathbf{Z}^{l-1}}(\mathbf{f}^{l-1}_{n,:})$ and $[\tilde{\boldsymbol{\Sigma}}^l]_{ij} = \Sigma_{\mathbf{S}^l,\mathbf{Z}^{l-1}}(\mathbf{f}^{l-1}_{i,:},\mathbf{f}^{l-1}_{j,:})$. The specific form of the functions $\mu_{\mathbf{m}^l,\mathbf{Z}^{l-1}}$ and $\Sigma_{\mathbf{S}^l,\mathbf{Z}^{l-1}}$ can be found in [30, Eqs. (7-8)]. Notice that we are able to compute the n-th marginal at each layer $\mathcal{N}(\mathbf{f}^l_{n,:}|[\tilde{\boldsymbol{\mu}}^l]_n,[\tilde{\boldsymbol{\Sigma}}^l]_{nn})$ since it only depends on the corresponding n-th input of the previous layer. So taking a sample of $\mathrm{q}(\mathbf{f}^L_{n,:})$ is straightforward, we have to recursively sample from the first to the last layer $\hat{\mathbf{f}}^1_{n,:} \to \hat{\mathbf{f}}^2_{n,:} \to \cdots \to \hat{\mathbf{f}}^L_{n,:}$. Specifically, we first sample from $\boldsymbol{\varepsilon}^l_n \sim \mathcal{N}(\mathbf{0},\mathbf{I}_{D^l})$ and then for $l = 1,\dots,L$ we sample:

$$\hat{\mathbf{f}}^l_{n,:} = \mu_{\mathbf{m}^l,\mathbf{Z}^{l-1}}(\hat{\mathbf{f}}^{l-1}_{n,:}) + \boldsymbol{\varepsilon}^l_n \cdot \sqrt{\Sigma_{\mathbf{S}^l,\mathbf{Z}^{l-1}}(\hat{\mathbf{f}}^{l-1}_{n,:},\hat{\mathbf{f}}^{l-1}_{n,:})}. \tag{A.22}$$

In summary, the expectation $\mathbb{E}_{\mathrm{q}(\mathbf{f}^L_{n,:})}[\log \mathrm{p}(y_n|\mathbf{f}^L_{n,:})]$ in the ELBO (see eq. (A.20)) can be approximated with a Monte Carlo sample generated using eq. (A.22). Since the ELBO factorizes across data points and the samples can be drawn independently for each point n, scalability is achieved through sub-sampling the data in mini-batches. The complexity to evaluate the ELBO and its gradients is $\mathcal{O}(N_b M^2 \sum_{l=1}^L D^l)$. Notice how the number of layers, and specifically the hidden dimension of each one, increases the computational cost in comparison to a single layer SVGP.

Once the ELBO is optimized, we can make predictions on an unseen test sample $\mathbf{x}_*$. The value of the latent variable $\mathbf{f}^L_{*,:}$ can be approximated by taking S samples[1] from the posterior up to the $(L-1)$-th layer using $\mathbf{x}_*$ as initial input. This yields a set $\{\mathbf{f}^{L-1}_{*,:}(s)\}_{s=1}^S$. Then, the density over $\mathbf{f}^L_{*,:}$ is given by the Gaussian mixture (recall that all the terms in eq. (A.21) are Gaussians):

$$\mathrm{q}(\mathbf{f}^L_{*,:}) \approx \frac{1}{S}\sum_{s=1}^S \mathrm{q}(\mathbf{f}^L_{*,:}|\mathbf{m}^L,\mathbf{S}^L;\mathbf{f}^{L-1}_{*,:}(s),\mathbf{Z}^{L-1}). \tag{A.23}$$

The code to perform DGP inference and prediction is integrated within GPflow (a GP framework built on top of Tensorflow) and is publicly available[2].

To illustrate the intuition behind DGPs, we show in figure A.13 samples from a 3-layer DGP on a 1D binary classification problem. We equipped each layer with an SE kernel and drew samples from the posterior distribution of the latent function at every layer. The SE kernel produces very smooth functions in the first layer. However, the concatenation of these simple functions produces more complex fuctions as we increase

[1] Results become stable after a few samples. Here, S was set to 100.

[2] https://github.com/ICL-SML/Doubly-Stochastic-DGP

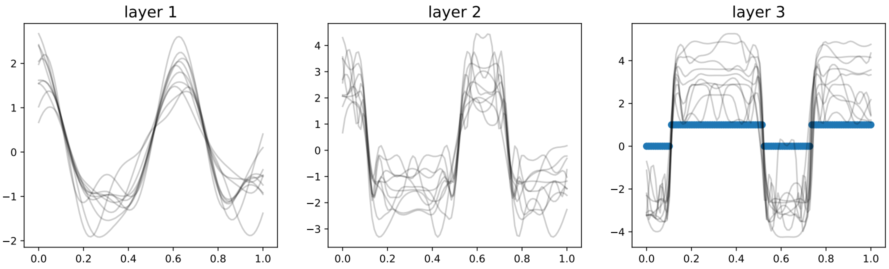


Figure A.13: Samples from the posterior distribution of the latent function at every layer of a 3-layer DGP on a 1D binary classification problem. Every layer is endowed with an SE kernel. The observations are described by the blue points on the third picture. Every layer provides a higher level of abstraction producing more complex patterns.

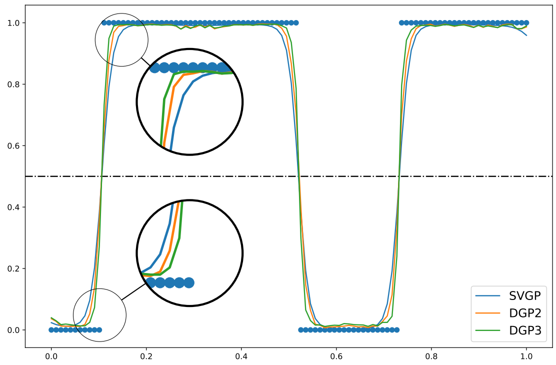


Figure A.14: Comparison of a SVGP, a 2-layer DGP (DGP2) and a 3-layer DGP (DGP3) on a 1D binary classification problem. Deeper models are able to capture better the decision boundary, see the zoomed-in areas.

the depth. In the last layer, it captures very sophisticated patterns combining flat regions with high-variability ones. These patterns can not be captured by a shallow GP with a stationary kernel. A comparison between a shallow SVGP and DGPs in this problem is shown in figure A.14. Both models perform very well because the problem is very simple, however, we can still notice one of the main differences between SVGPs and DGPs. Deeper models are able to make an abrupter jump defining better the decision boundary, see the zoomed-in areas. In this case, the SVGP is more uncertain on the decision boundary. All this motivates the use of DGPs instead of SVGPs for problems that require the capture of complex patterns.

References

[1] J. Wassermann, IASPEI New manual of seismological observatory practice, Vol. 1, GeoForschungsZentrum Potsdam, 2002, Ch. 13: Volcano seismology, p. 42.

[2] B. Chouet, Volcano seismology, pure and applied geophysics 160 (3) (2003) 739–788. doi:10.1007/PL00012556.
URL https://doi.org/10.1007/PL00012556

[3] D. J. Lary, A. H. Alavi, A. H. Gandomi, A. L. Walker, Machine learning in geosciences and remote sensing, Geoscience Frontiers 7 (1) (2016) 3 – 10, special Issue: Progress of Machine Learning in Geosciences. doi:https://doi.org/10.1016/j.gsf.2015.07.003.
URL http://www.sciencedirect.com/science/article/pii/S1674987115000821

[4] Q. Kong, D. T. Trugman, Z. E. Ross, M. J. Bianco, B. J. Meade, P. Gerstoft, Machine Learning in Seismology: Turning Data into Insights, Seismological Research Letters 90 (1) (2018) 3–14. arXiv:https://pubs.geoscienceworld.org/srl/article-pdf/90/1/3/4603086/srl-2018259.1.pdf, doi:10.1785/0220180259.
URL https://doi.org/10.1785/0220180259

[5] K. J. Bergen, P. A. Johnson, M. V. de Hoop, G. C. Beroza, Machine learning for data-driven discovery in solid earth geoscience, Science 363 (6433) (2019). arXiv:https://science.sciencemag.org/content/363/6433/eaau0323.full.pdf, doi:10.1126/science.aau0323.
URL https://science.sciencemag.org/content/363/6433/eaau0323

[6] E. Del Pezzo, A. Esposito, F. Giudicepietro, M. Marinaro, M. Martini, S. Scarpetta, Discrimination of Earthquakes and Underwater Explosions Using Neural Networks, Bulletin of the Seismological Society of America 93 (1) (2003) 215–223. arXiv:https://pubs.geoscienceworld.org/bssa/article-pdf/93/1/215/2714338/215_ssa02005.pdf, doi:10.1785/0120020005.
URL https://doi.org/10.1785/0120020005

[7] Q. Kong, R. M. Allen, L. Schreier, Myshake: Initial observations from a global smartphone seismic network, Geophysical Research Letters 43 (18) (2016) 9588–9594. arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2016GL070955, doi:10.1002/2016GL070955.
URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2016GL070955

[8] M. Curilem, F. Huenupan, C. San Martin, G. Fuentealba, C. Cardona, L. Franco, G. Acuña, M. Chacón, Feature analysis for the classification of volcanic seismic events using support vector machines, in: Nature-Inspired Computation and Machine Learning, Vol. 8857, Lecture Notes in Computer Science, 2014, pp. 160–171. doi:10.1007/978-3-319-13650-9_15.

[9] F. Giacco, A. M. Esposito, S. Scarpetta, F. Giudicepietro, M. Marinaro, Support vector machines and mlp for automatic classification of seismic signals at stromboli volcano, in: Proceedings of the 2009 Conference on Neural Nets WIRN09: Proceedings of the 19th Italian Workshop on Neural Nets, Vietri Sul Mare, Salerno, Italy, May 28–30 2009, IOS Press, NLD, 2009, p. 116–123.

[10] G. Curilem, J. Vergara, G. Fuentealba, G. Acuña, M. Chacón, Classification of seismic signals at villarrica volcano (chile) using neural networks and genetic algorithms, Journal of Volcanology and Geothermal Research 180 (1) (2009) 1 – 8. doi:https://doi.org/10.1016/j.jvolgeores.2008.12.002.
URL http://www.sciencedirect.com/science/article/pii/S0377027308006355

[11] M. C. Benitez, J. Ramirez, J. C. Segura, J. M. Ibanez, J. Almendros, A. Garcia-Yeguas, G. Cortes, Continuous hmm-based seismic-event classification at deception island, antarctica, IEEE Transactions on Geoscience and Remote Sensing 45 (1) (2007) 138–146. doi:10.1109/TGRS.2006.882264.

[12] M. Beyreuther, J. Wassermann, Continuous earthquake detection and classification using discrete hidden markov models, Geophysical Journal International 175 (3) (2008) 1055–1066. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-246X.2008.03921.x, doi:10.1111/j.1365-246X.2008.03921.x.
URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-246X.2008.03921.x

[13] P. B. Dawson, M. C. Benítez, J. B. Lowenstern, B. A. Chouet, Identifying bubble collapse in a hydrothermal system using hidden markov models, Geophysical Research Letters 39 (1) (2012). arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2011GL049901, doi:10.1029/2011GL049901.
URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011GL049901

[14] G. Cortés, L. García, I. Álvarez, C. Benítez, Ángel de la Torre, J. Ibáñez, Parallel system architecture (psa): An efficient approach for automatic recognition of volcano-seismic events, Journal of Volcanology and Geothermal Research 271 (2014) 1 – 10. doi:https://doi.org/10.1016/j.jvolgeores.2013.07.004.
URL http://www.sciencedirect.com/science/article/pii/S0377027313002229

[15] G. E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Computation 18 (7) (2006) 1527–1554, pMID: 16764513. arXiv:https://doi.org/10.1162/neco.2006.18.7.1527, doi:10.1162/neco.2006.18.7.1527.
URL https://doi.org/10.1162/neco.2006.18.7.1527

[16] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (8) (2013) 1798–1828. `doi:10.1109/TPAMI.2013.50`.

[17] Z. E. Ross, M.-A. Meier, E. Hauksson, P wave arrival picking and first-motion polarity determination with deep learning, Journal of Geophysical Research: Solid Earth 123 (6) (2018) 5120–5129. `arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2017JB015251`, `doi:10.1029/2017JB015251`.
URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2017JB015251`

[18] Y. Wu, Y. Lin, Z. Zhou, D. C. Bolton, J. Liu, P. Johnson, Deepdetect: A cascaded region-based densely connected network for seismic event detection, IEEE Transactions on Geoscience and Remote Sensing 57 (1) (2019) 62–75. `doi:10.1109/TGRS.2018.2852302`.

[19] S. M. Mousavi, W. Zhu, W. Ellsworth, G. Beroza, Unsupervised clustering of seismic signals using deep convolutional autoencoders, IEEE Geoscience and Remote Sensing Letters 16 (11) (2019) 1693–1697. `doi:10.1109/LGRS.2019.2909218`.

[20] M. Titos, A. Bueno, L. García, C. Benítez, A deep neural networks approach to automatic recognition systems for volcano-seismic events, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (5) (2018) 1533–1544. `doi:10.1109/JSTARS.2018.2803198`.

[21] M. Titos, A. Bueno, L. García, M. C. Benítez, J. Ibañez, Detection and classification of continuous volcano-seismic signals with recurrent neural networks, IEEE Transactions on Geoscience and Remote Sensing 57 (4) (2019) 1936–1948. `doi:10.1109/TGRS.2018.2870202`.

[22] M. Titos, A. Bueno, L. García, C. Benítez, J. C. Segura, Classification of isolated volcano-seismic events based on inductive transfer learning, IEEE Geoscience and Remote Sensing Letters (2019) 1–5`doi:10.1109/LGRS.2019.2931063`.

[23] A. Bueno, C. Benítez, S. De Angelis, A. Díaz Moreno, J. M. Ibáñez, Volcano-seismic transfer learning and uncertainty quantification with bayesian neural networks, IEEE Transactions on Geoscience and Remote Sensing (2019) 1–11`doi:10.1109/TGRS.2019.2941494`.

[24] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2006.

[25] J. Lee, J. Sohl-dickstein, J. Pennington, R. Novak, S. Schoenholz, Y. Bahri, Deep neural networks as gaussian processes, in: International Conference on Learning Representations, 2018.
URL `https://openreview.net/forum?id=B1EA-M-0Z`

[26] N. Lawrence, Deep learning, pachinko, and james watt: Efficiency is the driver of uncertainty, `http://inverseprobability.com/2016/03/04/deep-learning-and-uncertainty` (2016).

[27] A. Kendall, Deep learning is not good enough, we need bayesian deep learning for safe ai, https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai (2017).

[28] Y. Gal, Uncertainty in deep learning, Ph.D. thesis, University of Cambridge (2016).

[29] A. Damianou, N. Lawrence, Deep Gaussian processes, in: Artificial Intelligence and Statistics, 2013, pp. 207–215.

[30] H. Salimbeni, M. Deisenroth, Doubly stochastic variational inference for deep gaussian processes, in: Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 4588–4599.

[31] D. Moore, S. Russell, Signal-based Bayesian Seismic Monitoring, in: A. Singh, J. Zhu (Eds.), Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Vol. 54 of Proceedings of Machine Learning Research, PMLR, Fort Lauderdale, FL, USA, 2017, pp. 1293–1301.
URL http://proceedings.mlr.press/v54/moore17a.html

[32] D. A. Moore, S. J. Russell, Gaussian process random fields, in: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2, NIPS'15, MIT Press, Cambridge, MA, USA, 2015, p. 3357–3365.

[33] M. Noori, H. Hassani, A. Javaherian, H. Amindavar, S. Torabi, Automatic fault detection in seismic data using gaussian process regression, Journal of Applied Geophysics 163 (2019) 117 – 131. doi:https://doi.org/10.1016/j.jappgeo.2019.02.018.
URL http://www.sciencedirect.com/science/article/pii/S0926985118301964

[34] J. Hensman, A. G. de G. Matthews, Z. Ghahramani, Scalable variational gaussian process classification, in: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015, 2015.
URL http://jmlr.org/proceedings/papers/v38/hensman15.html

[35] M. Palo, J. M. Ibáñez, M. Cisneros, M. Bretón, E. Del Pezzo, E. Ocaña, J. Orozco-Rojas, A. M. Posadas, Analysis of the seismic wavefield properties of volcanic explosions at Volcán de Colima, México: insights into the source mechanism, Geophysical Journal International 177 (3) (2009) 1383–1398. arXiv:https://academic.oup.com/gji/article-pdf/177/3/1383/6052946/177-3-1383.pdf, doi:10.1111/j.1365-246X.2009.04134.x.
URL https://doi.org/10.1111/j.1365-246X.2009.04134.x

[36] G. Cortés, R. Arámbula, L. Gutiérrez, C. Benítez, J. Ibáñez, P. Lesage, I. Alvarez, L. Garcia, Evaluating robustness of a hmm-based classification system of volcano-seismic events at colima and popocatepetl volcanoes, in: 2009 IEEE International Geoscience and Remote Sensing Symposium, Vol. 2, 2009, pp. II–1012–II–1015. doi:10.1109/IGARSS.2009.5418275.

Chapter 3

Automatic classification of histopathological images using Shallow and Deep Gaussian Processes

3.1 Publication details

Authors: Ángel E Esteban, Miguel López-Pérez, Adrián Colomer, María A Sales, Rafael Molina, Valery Naranjo.
Title: A New Optical Density Granulometry-Based Descriptor for the Classification of Prostate Histological Images Using Shallow and Deep Gaussian Processes.
Publication: Computer Methods and Programs in Biomedicine, vol. 178, 303-317, September 2019.
Status: Published.
Quality indices:

- Impact Factor (JCR 2019): 3.632.
- Rank: 16/108 (Q1) in Computer Science, Theory & Methods.

3.2 Main contributions

- We create and make public a database of annotated prostate cancer images from the Clinical Hospital of Valencia.
- We demonstrate the importance of extracting tailored features from the optical density space instead of RGB images. We also formulate new morphological-based handcrafted features for cancer detection based on granulometry.
- We introduce the use of shallow and Deep Gaussian Processes for prostate cancer

classification using morphological and texture features and assess the generalization capability on an external database.

- We provide a fast and automatic method for analyzing whole slide images of the prostate and discerning whether there is cancer or not and where.

A New Optical Density Granulometry-Based Descriptor for the Classification of Prostate Histological Images Using Shallow and Deep Gaussian Processes

Ángel E. Esteban[a], Miguel López-Pérez[b,*], Adrián Colomer[a], María A. Sales[c], Rafael Molina[b], Valery Naranjo[a]

[a] *Institute of Research and Innovation in Bioengineering, I3B, Polytechnic University of Valencia, Valencia, Spain*
[b] *Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain*
[c] *Anatomical Pathology Service, University Clinical Hospital of Valencia, Valencia, Spain*

Abstract

Background and objective:
Prostate cancer is one of the most common male tumors. The increasing use of whole slide digital scanners has led to an enormous interest in the application of machine learning techniques to histopathological image classification. Here we introduce a novel family of morphological descriptors which, extracted in the appropriate image space and combined with shallow and deep Gaussian process based classifiers, improves early prostate cancer diagnosis.
Method:
We decompose the acquired RGB image in its RGB and optical density hematoxylin and eosin components. Then, we define two novel granulometry-based descriptors which work in both, RGB and optical density, spaces but perform better when used on the latter. In this space they clearly encapsulate knowledge used by pathologists to identify cancer lesions. The obtained features become the inputs to shallow and deep Gaussian process classifiers which achieve an accurate prediction of cancer.
Results:
We have used a real and unique dataset. The dataset is composed of 60 Whole Slide Images. For a five fold cross validation, shallow and deep Gaussian Processes obtain area under ROC curve values higher than 0.98. They outperform current state of the art patch based shallow classifiers and are very competitive

[*] Corresponding author. The first two authors contributed equally.

This work was supported by the the Spanish Ministry of Economy and Competitiveness through project DPI2016-77869. The Titan V used for this research was donated by the NVIDIA Corporation.

Email addresses: ngeesav@i3b.upv.es (Ángel E. Esteban), mlopez@decsai.ugr.es (Miguel López-Pérez), adcogra@i3b.upv.es (Adrián Colomer), salesman@gva.es (María A. Sales), rms@decsai.ugr.es (Rafael Molina), vnaranjo@dcom.upv.es (Valery Naranjo)

to the best performing deep learning method. Models were also compared on 17 Whole Slide test Images using the FROC curve. With the cost of one false positive, the best performing method, the one layer Gaussian process, identifies 83.87% (sensitivity) of all annotated cancer in the Whole Slide Image. This result corroborates the quality of the extracted features, no more than a layer is needed to achieve excellent generalization results.
Conclusion:
Two new descriptors to extract morphological features from histological images have been proposed. They collect very relevant information for cancer detection. From these descriptors, shallow and deep Gaussian Processes are capable of extracting the complex structure of prostate histological images. The new space/descriptor/classifier paradigm outperforms state-of-art shallow classifiers. Furthermore, despite being much simpler, it is competitive to state-of-art CNN architectures both on the proposed SICAPv1 database and on an external database.

Keywords: Prostate cancer, Histopathological Images, Gaussian Processes, Variational Inference, Granulometries, Deep Gaussian Processes.

1. Introduction

According to the World Health Organization, prostate cancer is the most common non-cutaneous cancer in men [1]. A histological diagnosis of prostate cancer is almost always required prior to instituting therapy for any stage of the disease. Pathologists determine the grade of cancer based on the formation, disposition, and structure of the glands (nuclei, lumen, cytoplasm and stroma) in the tissue, scoring the samples between 1 to 5, following the Gleason grading system [2], see Figure 1.

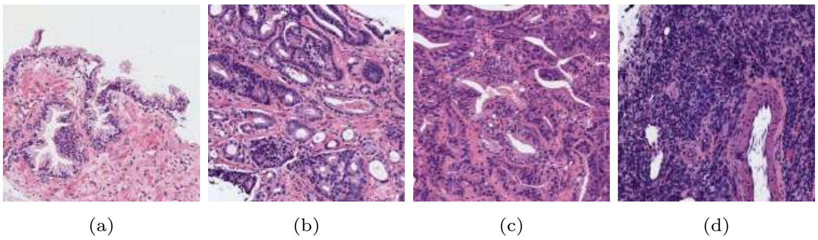

(a) (b) (c) (d)

Figure 1: Examples of Gleason grades of histological images: (a) benign; (b) grade 3; (c) grade 4; (d) grade 5.

Tissue histopathological slides can nowadays be acquired and digitally stored thanks to the advent of whole slide digital scanners. The widespread use of such scanners has led to an increasing interest on applying machine learning techniques to classify these images, for a review of this topic, see [3]. Due to the large resolution of the images obtained under the microscope, evaluating each single diagnostic test manually is a very time-consuming task. This fact encourages the research on CAD algorithms that decrease pathologists workload by recognizing obviously benign cases so that experts can focus on the delicate ones [4].

In digital brightfield microscopy, tissues are usually stained before digitization and evaluation by pathologists. Hematoxylin and Eosin (H&E) are probably the most widely used combination of stains. Since Color Deconvolution (CD), that is, H&E separation, is a very important preprocessing step, several methods have been developed (see [5] for a recent review). One of the first CD methods, which is widely used, was proposed by Ruifrok et al. [6]. This is a supervised method where the stain color vectors are obtained by measuring the relative absorption of each stain in single-stained images. These color vectors are used on all the WSI images to obtain their RGB and Optical Density (OD) space H&E images. CAD algorithms based on hand-driven approaches use RGB space H&E images, while deep learning approaches work directly with the orignal RGB images. In this paper we will show that the selection of the space where H&E are represented significantly affects the performance of classifiers.

Two approaches are currently being used in the literature to detect tumorous prostatic tissues. One is based on segmenting the images and identifying the regions of interest (ROIs), while the other utilizes patches for classification purposes. In this work, we follow the second approach: the entire whole slide image (WSI) is split into patches and each one is analyzed independently. While pathologists use several scales (magnification factors), most machine learning algorithms use a single one. Gupta et al. [7] compare different scales for training and test in breast histology. They conclude that with suitable features together with an ensemble classifier framework, such as bagging or boosting, the classification can be made largely magnification invariant. For a selected magnification factor and patch size, a feature extraction process to encode the relevant information of the images must be carried out.

Nowadays, the remarkable progress in the deep learning field allows to automatically compute high-abstraction feature maps by means of neural networks based on stacks of convolutional blocks (a.k.a. convolutional neural networks or CNNs). CNNs are being successfully applied in many computer vision tasks. In the particular case of histological images, CNNs have also benefit of the automatic feature extraction for the classification of different tumoral patterns in diverse organs [8]. Le Hou et al. [9] use a CNN for path-based classification which achieves good results discriminating different cancer subtypes in WSIs. The BACH challenge[1] resulted in several works [10, 11, 12] in which the different types of breast cancer including in-situ carcinoma, invasive tumor, and benign tumor were automatically identified by means of well-known CNN architectures: Inception v3, Xception and ResNet. A fine-tuning process of the same architectures was carried out by Ferlaino et al. [13] to robustly localize and classify placental cells using histological images. Shallu et al. [14] demonstrated

[1] `https://iciar2018-challenge.grand-challenge.org/`

that transfer learning is better than training from scratch in breast cancer histological image classification, obtaining very good accuracy with the VGG16 and VGG19 architectures. In prostate cancer histology, CNNs have recently been utilized for semantic segmentation grading [15, 16]. These methods provide for each pixel its probability of belonging to each class.

According to Komura et al. [3], the relevant information to classify histological images is related to texture and morphology. Although CNNs are able to learn these feature representations, textural and morphological tissue properties can also be manually captured by a suitable hand-crafted descriptor avoiding specific hardware requirements and reducing computational cost. Therefore, the information (descriptors) extracted from each patch becomes the key to a successful tissue classification. Generic descriptors, such as HOG [17], LBP [18], SIFT [19] or Gabor filters [20] are frequently used for prostate cancer detection. Kumar et al. [21] show that LBP are as good as deep features and dictionaries with the benefits of easy computation and low dimensionality. Recent works in the field [22, 23, 24, 25] also indicate that descriptors based on structural and morphological properties of the prostatic tissue could outperform those based on standard features. It is also possible to combine a convolutional neural network with handcrafted features as Zhou et al. [26] but it is not widely used in the literature.

In a hand-driven learning paradigm, once a descriptor has been selected, a suitable classifier must be chosen. Although ensemble classifiers as Random Forests [27], Adaboost [19] or Xgboost [28] have been used, it could be said that Support Vector Machine (SVM) is the preferred classifier [23, 29, 30]. Unfortunately, nonparametric probabilistic models which take into account the uncertainty of the predictions, particularly Gaussian Processes (GPs) [31], which are in the state-of-art in classification, have been less used. It has long been known that neural networks with an *infinite* width are equivalent to Gaussian Processes with a certain covariance kernel. GPs have the advantage of been nonparametric, unlike neural networks that have to learn a large number of parameters in order to have a sufficiently complex model. GPs allow us to use a sound framework with a well defined inference procedure. Prior models in the form of different kernels can be used to encapsulate knowledge on the problem at hand. Model parameters can be automatically estimated without hand-tuning and predictions go beyond point estimates to provide very important information on uncertainty. They are starting to be used in histological image classification. Kandemir et al. [32] proposed a multi-instance relational learning based on GPs for histhopathology images. For the multi-instance purpose, they process each image as a bag and each patch as an instance. In order to capture the differences in cell formations caused by the disease status, they also introduce relational learning between instances and add relational side information from the spatial positions of segmented cells. More recently, with the purpose of facing more complex models, Deep Gaussian Processes (DGPs) [33] have been proposed. Unlike deep learning that requires a large dataset to learn a good model, DGPs can be applied with success even when data is scarce. In the last years, the ML community has experienced a remarkable interest in DGPs which are a hierarchical extension of GPs. Roughly speaking, they are deep architectures (like CNNs) whose layers are modelled by probabilistic GPs. This brings all the advantages of using GPs and provides much more power to approximate complex patterns in data. Results are really promising, surpassing CNNs in several problems. Unfortunately, in spite of its representation power, there are hardly any works in histopathology that make use of DGPs, see, however, Kandemir et al. [34] who apply a two-layer DGP model in histopathology cancer classification using an asymmetric transfer learning approach. The dataset used was built from two different tissues: breast and esophagus.

Once a patch classifier has been learned (using either hand-crafted or learned features), an image level evaluation is needed for prostate cancer diagnosis. Some works utilize a multiple instance learning approach and provide an overall WSI diagnosis, see Campanella et al. [35]. Another approach, which is frequently followed, is presented in Litjens et al. [8]. For each pixel, the probability of being cancerous is estimated from the patch probabilities, constructing a heat map for the WSI. This probability map is then thresholded to classify every WSI pixel as cancerous or benign.

In this work we approach the classification of prostate histological images by first calculating the OD of each WSI to then estimate its H&E concentration components (we will show that OD is a better space than RGB for feature extraction and classification tasks). Hand-crafted features, which are expected to capture the expertise of pathologists, are then extracted from patches of these two concentration components. Finally, patches are classified using single-layer and multilayer Gaussian processes into benign and cancerous classes. We also carry out a validation at WSI level. We predict the per pixel probability of being cancerous and validate the obtained probability map. GPs and DGPs perform similarly and they are competitive to the tested shallow and deep classifiers. In other words, the quality of our OD extracted features does not require more than a single-layer GP to outperform the best performing classifiers.

The rest of the paper is organized as follows, in section 2 we introduce and describe a new WSI database of histological prostate images which has been manually annotated by experts. In section 3, we explain how the CD task is performed on each WSI and describe how to obtain its RGB and OD H&E representations. In section 4, we motivate and define our new two morphological descriptors, we explain how the proposed framework, to discriminate between cancer and benign tissue in prostate, tries to mimic the way of analysis of a pathologist. In section 5, we provide an introduction to GPs and its hierarchical extension, DGP, in supervised learning. In section 6 we carry out a comparative study of several classifiers using the proposed features in a real clinical database provided by pathologists from the Hospital Clínico of Valencia. The performed experiments show that the classifier based on GP and deep GP together with the proposed features extracted in the OD space outperforms the current state of the art shallow classifiers and it is competitive to state-of-art deep convolutional neural netwok classifiers. In the experimental discussion we provide an insightful analysis. We use the area under the curve (AUC) for the evaluation of patch classification and FROC for diagnosis (detection) of prostate cancer in Whole

Slide Image. We also analyze its complexity and computational cost compared to CNNs. Besides, to assess the robustness, we use the database proposed in [15, 29] for external validation. Finally, in Section 7 we summarize the conclusions extracted from our experimental results.

2. Material: SICAP database

The lack of large and public databases of prostate histopathological images has prevented researchers from a rigorous and meaningful comparison of supervised learning methods on these images. To the best of our knowledge, only three public databases containing histological prostate images are available. The first one, which is the result of a joint work by the National Cancer Institute and the National Human Genome Research Institute, both from United States, has generated comprehensive, multi-dimensional maps of the key genomic changes in 33 types of cancer. However, the fact of not providing pixel-wise annotations along with a large amount of missing labels makes this database [2] inappropriate to validate new methodologies. The second one, the public database released by the authors of [36], is composed by 886 images and their corresponding pixel-wise annotations according to the Gleason scale. Unfortunately, only isolated tissue spots, representing characteristic patterns, are provided which prevents a patch size comparison and a full WSI classification. The third one, a database used in [15, 29] is composed by 625 different grade patches with a pixel-wise mask provided by pathologists. No WSIs are provided.

In this work, we present the SICAPv1 database, publicly available at `https://cvblab.synology.me/PublicDatabases/SICAPv1.zip`. It was obtained by a team of pathologists working at the Hospital Clínico of Valencia. Biopsies of 48 different patients were processed, hematoxylin and eosin stained and then digitized using the *Ventana iScan Coreo* scanner at 40x magnification. The database consists of 79 WSI: 19 correspond to benign prostate tissue biopsies (negative class) and 60 to pathological prostate tissue biopsies (positive class). Note that the entire dataset was divided into two subsets, 60 WSI (17 benign and 43 pathological) were used to learn the models and the remaining 19 images (two benign, seven diagnosed as grade 3, eight corresponding to grade 4 and two grade 5 WSIs) to test them. The malignant regions of the pathological images were carefully pixel-wise annotated by an expert team of pathologists. For this purpose, experts manually annotated the relevant tumoral areas using an online in-house application based on the OpenSeadragon functional core [37].

In order to automatically analyse these gigapixel images, the images were downsampled from 40× to 10× and divided in patches with a 50% overlap. To test the influence of the patch size, different sizes were selected: 512^2 and 1024^2, resulting on the two different datasets detailed in Table 1. Note that malignant patches were extracted from the annotated tumoral areas in the positive class

[2] `https://portal.gdc.cancer.gov/`

Table 1: SICAPv1 database description. Number of training WSIs and number of $512^2/1024^2$ associated patches.

	Benign	Grade 3	Grade 4	Grade 5	Pathological
#WSIs	17	18	15	10	43
#512^2 patches	6725	380	589	173	1142
#1024^2 patches	1909	113	181	50	344

images. Patches less than 25% inside a malignant area were not considered. And benign patches were extracted from benign WSIs.

3. Color deconvolution

For each WSI, the three-channel image information is the RGB intensity detected by a brightfield microscope observing a stained prostate histological slide. H&E are the stains usually used in pathology: Hematoxylin highlights the nuclei in purple and Eosin the stroma and cytoplasm in pink. Each $M \times N$ image is denoted by $\mathbf{I}$ with columns $\mathbf{i}_c = (i_{1c}, \ldots, i_{MN_c})^T,\ c \in \{R, G, B\}$.

We follow the color deconvolution approach described in [6]. According to the Lambert-Beer's law we can express the OD for channel c of the slide as $\mathbf{y}_c = -\log(\mathbf{i}_c/\mathbf{i}_c^0) \in \mathbb{R}^{MN\times1}$, where $i_c^0 = 255$ is the incident light and division inside the logarithm is performed element-wise. Slides are stained using $\mathbf{n}_s = 3$ stains, $s \in \{H, E, Res\}$ (to obtain a unique stain decomposition we consider a third stain which represents the residual part) then the observed OD multichannel $\mathbf{Y} = [\mathbf{y}_R, \mathbf{y}_G, \mathbf{y}_B] \in \mathbb{R}^{MN\times3}$ can be decomposed as a matrix multiplication $\mathbf{Y}^T = \mathbf{M}\mathbf{C}^T$, where $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3] \in \mathbb{R}^{MN\times3}$ is the stain concentration matrix, with $\mathbf{c}_s$, the s-th column of $\mathbf{C}$, containing at each pixel position the concentration of stain color s and $\mathbf{M} \in \mathbb{R}^{3\times3}$ denoting the normalized stain matrix of the fixed form exposed in [6]. Notice that the s-th column of $\mathbf{M}$, $\mathbf{m}_s$, denotes the specific color of stain s.

The stain concentration matrix can then be recovered using $\mathbf{C}^T = \mathbf{M}^{-1}\mathbf{Y}^T$. Concentrations are transformed back to color (RGB) images using $\mathbf{y}_s^{sep} = \exp(-\mathbf{m}_s\mathbf{c}_s^T), s \in \{H, E\}$. Features are usually extracted from the single channel images $\exp(-\mathbf{c}_s), s \in \{H, E\}$ in the so called RGB space. In this work, we propose to perform this step in the OD space where stains are linearly separable, that is, directly on $\mathbf{c}_s, s \in \{H, E\}$. Figure 2 shows three different images from three different biopsies (and patients), one benign and two pathological, and their corresponding OD concentrations, Hematoxylin in the first row and Eosin in the second one. OD Hematoxylin captures nuclei infomation while OD Eosin contains information on stroma and cytoplasms.

4. Granulometry-based descriptors

Granulometry is a technique based on mathematical morphology. Size distributions of different elements in an image are obtained applying a series of

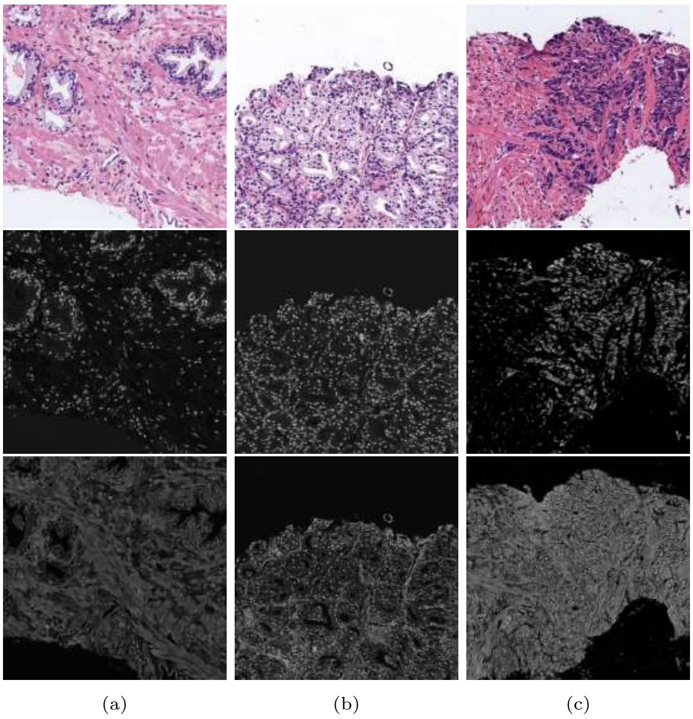


Figure 2: Hematoxylin (second row) and Eosin (third row) optical densities for three samples: a) Benign; b) and c) pathological.

morphological opening (or closing) operations with increasing-size structuring elements. The obtained size distribution provides shape and size information. In this paper, we propose the use of the classic formulation of granulometry as a new descriptor used in histhological images and define a new variant for prostate cancer classification which makes use of morphological reconstruction. The two proposed descriptors are explained below.

4.1. Granulometry-based descriptor

Based on a pyramid of morphological operators, granulometry calculates the size distribution of bright and dark objects present in an image. Let $\mathbf{z}$ be either a whole gray level image or an image patch. We can define a morphological descriptor, using the opening operator $\gamma_i(\mathbf{z})$ applied to the image $\mathbf{z}$ with a SE (window) of size i. This opening operator can be expressed as the combination of an erosion ($\epsilon_i(\mathbf{z})$) followed by a dilation ($\delta_i(\mathbf{z})$), both with the SE of size i. When this opening is computed with a SE of increasing size (λ), we obtain a morphological opening pyramid (or granulometry profile) which can be formalized as:

$$\Pi_\gamma(\mathbf{z}) = \{\Pi_{\gamma\lambda} : \Pi_{\gamma\lambda} = \gamma_\lambda(\mathbf{z}), \forall\lambda \in [0, s, 2s, ..., n_{max}]\}. \tag{1}$$

where n_{max} represents the maximum size of the structuring element, and the sizes increase in steps s.

Making use of the opening pyramid (Π_γ), the granulometry curve or pattern spectrum of $\mathbf{z}$, $PS_\Gamma(\mathbf{z}, n)$, can be defined as:

$$PS_\Gamma(\mathbf{z}, n) = \frac{m(\Pi_{\gamma n}(\mathbf{z})) - m(\Pi_{\gamma n+1}(\mathbf{z}))}{m(\mathbf{z})},\ n \geq 0 \tag{2}$$

where $m(\mathbf{z})$ is the Lebesgue measure of $\mathbf{z}$ and it is computed as the area of $\mathbf{z}$ in the binary case and the volume in the gray-scale case (sum of pixel values).

$PS_\Gamma(\mathbf{z}, n)$ (also called size density of $\mathbf{z}$) maps each size n to a measure of the bright image structures with this size: loss of bright image structures between two successive openings. It is a probability density function (a histogram) in which a large impulse in the pattern spectrum at a given scale indicates the presence of many image structures at that scale.

By duality, a closing, $\varphi_i(\mathbf{z})$ is defined as the dilation of $\mathbf{z}$ followed by an erosion, both with a SE of size i. In the same way, a morphological closing pyramid is an anti-granulometry profile and can be computed on the image performing repeated closings with a SE of increasing size (λ) defined as:

$$\Pi_\varphi(\mathbf{z}) = \{\Pi_{\varphi\lambda} : \Pi_{\varphi_\lambda} = \varphi_\lambda(\mathbf{z}), \forall\lambda \in [0, ..., n_{max}]\} \tag{3}$$

The concept of pattern spectrum extends to the anti-granulometry curve $PS_\Phi(\mathbf{z})$ with respect to the family of closings Φ:

$$PS_\Phi(\mathbf{z}, -n) = \frac{m(\Pi_{\varphi n}(\mathbf{z})) - m(\Pi_{\varphi n-1}(\mathbf{z}))}{m(\mathbf{z})},\ n \geq 0. \tag{4}$$

Notice that this spectrum characterises the size of image structures with low level intensities.

Both granulometry and anti-granulometry descriptors are concatenated to construct the final descriptor (*Gran*).

4.2. Geodesic Granulometry-based descriptor

In this work, we introduce a variant of the granulometry, named geodesic granulometry, which is based on geodesic transformations.

A geodesic transformation involves two images: a marker image (or patch) $\mathbf{y}$ and a reference image $\mathbf{z}$. The *geodesic dilation* is the iterative unitary dilation of $\mathbf{z}$ with respect to $\mathbf{y}$, that is:

$$\delta_{\mathbf{y}}^{(n)}(\mathbf{z}) = \delta_{\mathbf{y}}^{(1)}\delta_{\mathbf{y}}^{(n-1)}(\mathbf{z}),\ \text{being}\ \delta_{\mathbf{y}}^{(1)}(\mathbf{z}) = \delta_B(\mathbf{z}) \wedge \mathbf{y}. \tag{5}$$

The *reconstruction by dilation* is the successive geodesic dilation of $\mathbf{z}$ regarding $\mathbf{y}$ up to idempotence, that is:

$$R_{\mathbf{y}}^{\delta}(\mathbf{z}) = \delta_{\mathbf{y}}^{(i)}(\mathbf{z}),\ \text{so that}\ \delta_{\mathbf{y}}^{(i)}(\mathbf{z}) = \delta_{\mathbf{y}}^{(i+1)}(\mathbf{z}). \tag{6}$$

The *reconstruction by erosion* can be obtained as its dual operator:

$$R_{\mathbf{y}}^{\varepsilon}(\mathbf{z}) = [R_{\mathbf{y}^c}^{\delta}(\mathbf{z}^c)]^c, \tag{7}$$

being $\mathbf{z}^c$ the complement image (or patch).

The reconstruction by dilation removes from the reference $\mathbf{z}$ the bright objects unconnected with the marker $\mathbf{y}$. The underlying idea on which the new descriptor is based is to only consider in the granulometry spectrum the objects totally removed in each opening (closing) step. Using $\gamma(\mathbf{z})$ as indicated in Equation (1) can lead to the inclusion in the pattern spectrum of fragments of objects partially removed in the process. To solve this shortcoming, we modify the granulometry profile (Equation (1)) by using the geodesic opening given by $\gamma^r(\mathbf{z}) = R^{\delta}_{\gamma(\mathbf{z})}(\mathbf{z})$. By duality, the proposed geodesic closing, to be used in the computation of the anti-granulometry profile, (Equation (3)) is $\varphi^r(\mathbf{z}) = R^{\varepsilon}_{\varphi(\mathbf{z})}(\mathbf{z})$. The new geodesic granulometry descriptors will be denoted $PS^r_{\Gamma}(\mathbf{z}, n)$ and $PS^r_{\Phi}(\mathbf{z}, -n)$, respectively.

Both geodesic descriptors are concatenated to construct the final descriptor (*GeoGran*).

4.3. Granulometry profiles for prostate cancer detection

The proposed framework, to discriminate between cancer and benign tissue in prostate, tries to mimic the way of analysis of a pathologist. Basically, the cancer destroys the tissue structure. A benign tissue is formed by glands, each of them with a lumen surrounded by cytoplasm and nuclei, distributed in a background of stroma (which also contains sparsely distributed nuclei) (Figure 2(a)). As cancer progresses, glands begin to proliferate and merge, destroying the structure of benign tissues. Cytoplasm and lumens disappear and stroma is invaded by nucleis. Figure 2, (first row), shows three different cancer stages ((a) benign, (b) grade 3, (c) grade 5). To capture in a descriptor the tissue structure, we propose to use PS_{Φ} with H as input image. This encodes the structure of the glands by recovering the structure of the nuclei which formed the gland frontiers (those that enclosed their lumen and cytoplasm). The granulometric profiles, Π_{φ}, for the three image examples are shown in Figures 3(c), 4(c) and 5(c). To capture stroma information, PS_{Γ} is applied on the E component. Figures 3(a), 4(a) and 5(a) show the Π_{γ} profiles for the three examples. Figures 3, 4 and 5 also depict in columns (b) and (d) the geodesic profiles Π^r_{γ} and Π^r_{φ}, respectively. Note that Π^r_{φ} (columns (d)), for the three cases, shows that the results for different steps (different sizes of SEs) of the granulometric profile do not change. This suggests that stroma information more accurately extracted in PS^r_{Γ}, is the most relevant information to discriminate between pathological and benign tissues (as results presented in the experimental section corroborate).

5. Probabilistic model and inference

In this section we provide a brief introduction to the use of GPs and DGPs in supervised learning. An in depth study of these models can be found in [31] and [38]. Let us assume that we have n labeled training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where $\mathbf{x}_i \in \mathbb{R}^d$ is the feature vector, $y_i \in \{0, 1\}$ for a binary classification problem, and $y_i \in \mathbb{R}$ for a regression one. We use either $y_i = f_i + \epsilon_i$ or

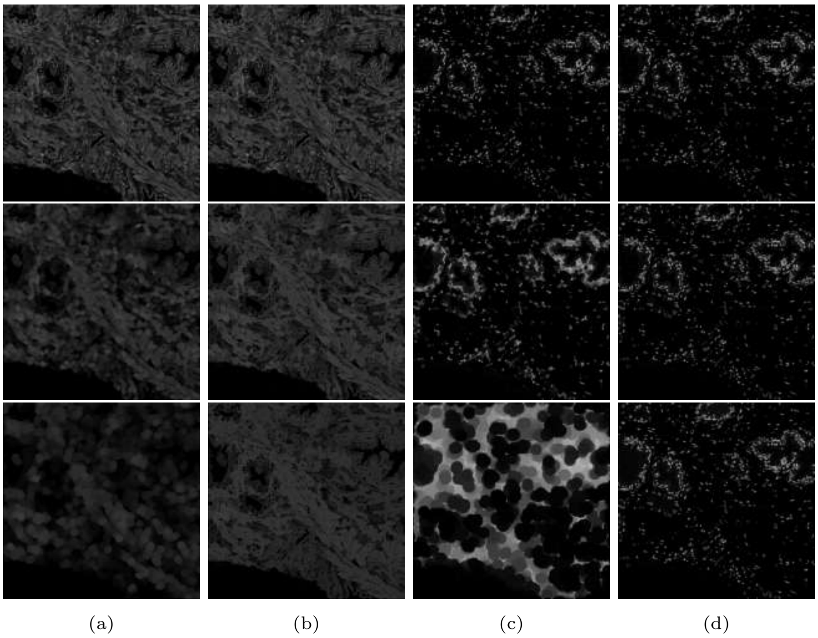


Figure 3: Granulometry profiles (steps $s = 1, 4, 16$) for image (a) in Figure 2: (a) Π_{φ}; (b) Π^{r}_{φ}; (c) Π_{γ}; (d) Π^{r}_{γ}.

$\mathrm{p}(y_i|f_i) = \sigma^{y_i}(f_i)\sigma^{1-y_i}(f_i)$ depending on whether we are dealing with a regression or classification problem, respectively. We assume that the noise in the regression problem is uncorrelated Gaussian of variance ρ^2 and $\sigma(\cdot)$ denotes the sigmoid function. We have used f_i instead of $f(\mathbf{x}_i)$ for simplicity. Notice that to tackle both problems we need to model the behavior of the function $f(\cdot)$ on seen and unseen samples $\mathbf{x}$.

5.1. Single-layer Gaussian Process

In a GP based formulation of a supervised problem we assume that the distribution of $\mathbf{f} = (f_1, \ldots, f_n)^{\mathrm{T}}$ given $\mathbf{X}$ is a multivariate normal, $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where the zero mean is assumed for simplicity and $\sigma_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. where $k(\cdot, \cdot)$ is a kernel function. The use of kernel functions will guarantee that $\mathbf{\Sigma}$ is always a semidefinite positive matrix (independently of the number of samples and the features in $\mathbf{X}$). In this paper we use the squared exponential kernel (SE), also known as Radial Basis Function (RBF), defined as:

$$k(\mathbf{x}, \mathbf{x}') = C \exp(-\gamma ||\mathbf{x} - \mathbf{x}'||^2). \tag{8}$$

where the parameters C and γ will be estimated from the observations (the learning task).

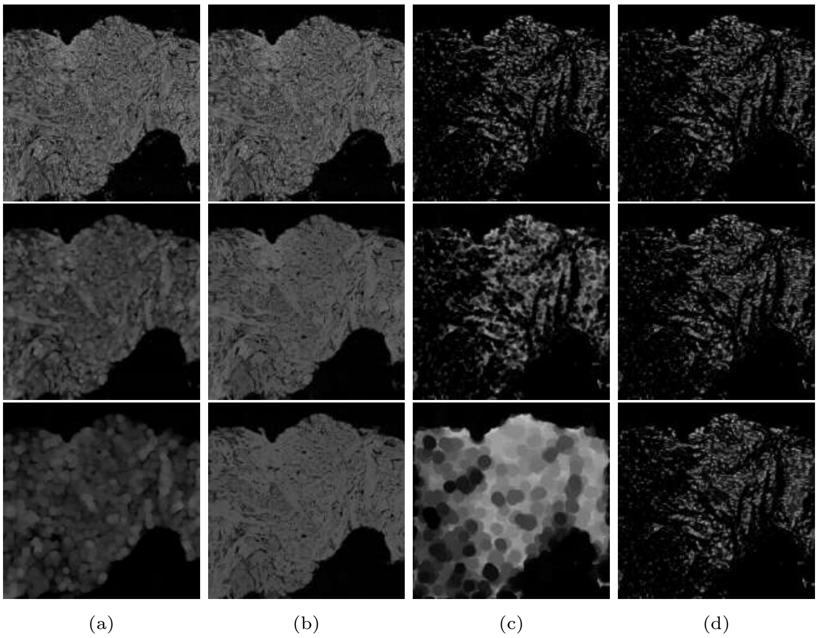


Figure 4: Granulometry profiles (steps $s = 1, 4, 16$) for image (b) in Figure 2: (a) Π_φ; (b) Π^r_φ; (c) Π_γ; (d) Π^r_γ.

Now we have all the ingredients we need to model our supervised learning problem using GPs. Given $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$ we write

$$\mathrm{p}(\mathbf{y}, \mathbf{f}) = \prod_{i=1}^{n} \mathrm{p}(y_i|f_i)\mathrm{p}(\mathbf{f}|\mathbf{X}) \tag{9}$$

and proceed with the learning and inference tasks. We first learn the model parameters (C, γ and for a regression problem ρ^2 as well) by maximizing on them the marginal log-likelihood, that is,

$$\log \mathrm{p}(\mathbf{y}) = \log \int \mathrm{p}(\mathbf{y}|\mathbf{f})\mathrm{p}(\mathbf{f}|\mathbf{X})\mathrm{d}\mathbf{f} \tag{10}$$

which will allow us to calculate $\mathrm{p}(\mathbf{f}|\mathbf{y})$ and finally perform inference: given a new feature vector $\mathbf{x}^*$, we calculate

$$\mathrm{p}(f_*|\mathbf{y}, \mathbf{x}_*, \mathbf{X}) = \int \mathrm{p}(f_*|X, \mathbf{x}_*, \mathbf{f})\mathrm{p}(\mathbf{f}|X, \mathbf{y})d\mathbf{f} \tag{11}$$

which will allow us to predict $\mathbf{y}_{\mathbf{x}^*}$. There are two problems that must be faced when using GP in supervised learning. The first one, which is easier to handle, comes from the fact that in classification problems the prior distribution is not conjugate for the observation model. That is usually handled by maximizing

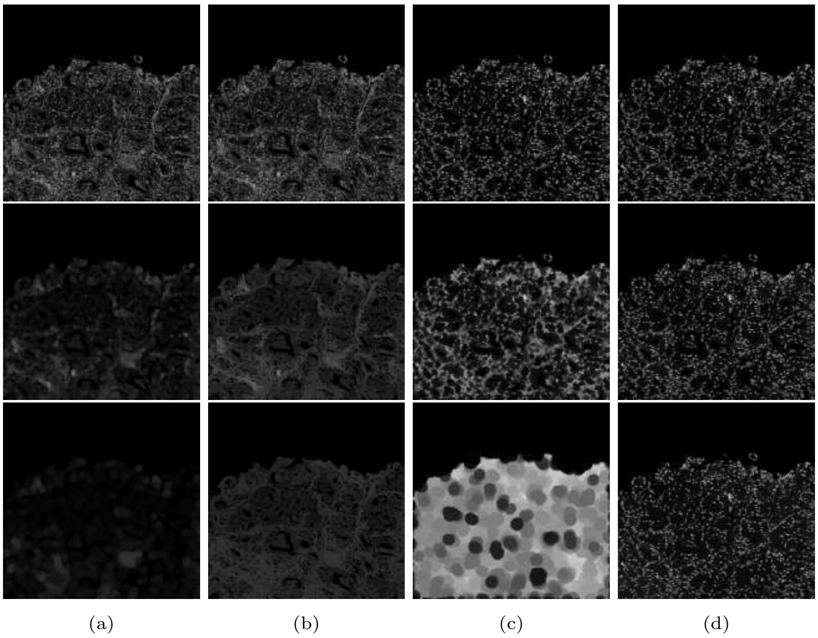


Figure 5: Granulometry profiles (steps $s = 1, 4, 16$) for image (c) in Figure 2: (a) Π_{φ}; (b) Π^{r}_{φ}; (c) Π_{γ}; (d) Π^{r}_{γ}.

a lower bound of the marginal likelihood in eq. 10. This will also have the effect of obtaining an approximation to $\mathrm{p}(\mathbf{f}|\mathbf{y})$ but not the real one, however, this problem is less relevant than the second one. Maximizing eq. 10 requires inverting a matrix the size of the number of samples (an $\mathcal{O}(n^3)$ operation) which is prohibitive for large datasets.

The most popular approach to dealing with the computational burden of GPs is to introduce $m \ll n$ *inducing points* $\mathbf{u} = (u_1, \ldots, u_m)$ which the inference is based on. These are GP realizations at the *inducing locations* $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\} \subset \mathbb{R}^d$, just like $\mathbf{f}$ is at the inputs $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ [39], in other words, $\mathbf{u} = f(\mathbf{Z})$. We can rewrite the joint distribution as

$$\mathrm{p}(\mathbf{y}, \mathbf{f}, \mathbf{u}) = \underbrace{\prod_{i=1}^{N} \mathrm{p}(y_i|f_i)}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathrm{p}(\mathbf{u}; \mathbf{Z})}_{\text{GP prior}} \tag{12}$$

where a semicolon is used to specify the inputs of the GP, this will clarify multilayer-models notation.

Notice that we have overloaded the notation a bit to make clear the introduction of the inducing points but no changes in the modelling have been introduced since $\mathrm{p}(\mathbf{f}) = \int \mathrm{p}(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathrm{p}(\mathbf{u}; \mathbf{Z})\mathrm{d}\mathbf{f}$.

Equipped with this decomposition, we go back to the marginal likelihood function in eq. 10 and use Jensen's inequality to, following the approach in [40], write

$$\log \mathrm{p}(\mathbf{y}) \geq \int \mathrm{q}(\mathbf{u})\mathrm{p}(\mathbf{f}|\mathbf{u};\mathbf{Z}) \log \frac{\mathrm{p}(\mathbf{y}|\mathbf{f})\mathrm{p}(\mathbf{f}|\mathbf{u};\mathbf{X},\mathbf{Z})\mathrm{p}(\mathbf{u};\mathbf{Z})}{\mathrm{p}(\mathbf{f}|\mathbf{u};\mathbf{X},\mathbf{Z})\mathrm{q}(\mathbf{u})} \mathrm{d}\mathbf{u}\mathrm{d}\mathbf{f}. \tag{13}$$

Now the optimization process becomes more involved. We have to estimate, together with the model parameters (C, γ and for a regression problem ρ^2 as well), the parameters of the distribution $\mathrm{q}(\mathbf{u})$ which is usually assumed to be a multivariate Gaussian, and the inducing point locations $\mathbf{Z}$. The benefit is that this learning process has become $\mathcal{O}(nm^2)$. Finally, $\mathrm{q}(\mathbf{u})$ is used, instead of $\mathrm{p}(\mathbf{f}|\mathbf{y})$, in eq. 11 for the inference (testing) process.

5.2. Deep Gaussian Processes

In standard (single-layer) GPs, the output of the GP is directly used to model the observed response $\mathbf{y}$. However, this output could be used to define the input locations of another GP. If this is repeated L times, we obtain a hierarchy of GPs that is known as a Deep Gaussian Process (DGP) with $L+1$ layers. DGPs were first introduced in [33], they can be used for regression and classification problems by placing appropriate likelihoods (like the ones introduced at the beginning of this section) after the last layer.

Unfortunately, exact inference in DGP is intractable (beyond the the computationally expensiveness of GPs and the non-conjugacy of the prior), as it involves integrating out latent variables that are used as inputs in the next layer (i.e. they appear inside a complex kernel matrix). To overcome this, again m inducing points $\mathbf{u}^l$ at inducing locations $\mathbf{z}^{l-1}$ are introduced at each layer l. We write the joint distribution of the observation and DGP as

$$\mathrm{p}(\mathbf{y}, \{\mathbf{f}, \mathbf{u}^l\}_{l=1}^L) = \underbrace{\prod_{i=1}^N \mathrm{p}(y_i|f_i^L)}_{\text{likelihood}} \underbrace{\prod_{l=1}^L \mathrm{p}(\mathbf{f}^l|\mathbf{u}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1})\mathrm{p}(\mathbf{u}^l; \mathbf{z}^{l-1})}_{\text{DGP prior}}. \tag{14}$$

Here, $\mathbf{f}^0 = \mathbf{X}$, and each factor in the product is the joint distribution over $(\mathbf{f}^l, \mathbf{u}^l)$ of a GP in the inputs $(\mathbf{f}^{l-1}, \mathbf{z}^{l-1})$, but rewritten with the conditional probability given $\mathbf{u}^l$. For notation simplicity, in this description the dimension of the hidden layers has been fixed to one. This can be generalized straightforwardly, in this case $\mathbf{f}^l, \mathbf{u}^l$ and $\mathbf{z}^{l-1}, l = 1, \ldots, L$ will be matrices of the appropriate sizes, see [33, 38].

To train the model, we follow the approach in [38] where the authors use the Jensen's inequality, with the posterior distribution approximation

$$\mathrm{q}(\{\mathbf{f}^l, \mathbf{u}^l\}_{l=1}^L) = \prod_{l=1}^L \mathrm{p}(\mathbf{f}^l|\mathbf{u}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1})\mathrm{q}(\mathbf{u}^l). \tag{15}$$

where $\mathrm{q}(\mathbf{u}^l) = \mathcal{N}(\mathbf{u}^l|\mathbf{m}^l, \mathbf{S}^l)$, to write

$$\begin{aligned}\log \mathrm{p}(\mathbf{y}) \geq \int \prod_{l=1}^{L} \mathrm{p}(\mathbf{f}^l|\mathbf{u}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1})\mathrm{q}(\mathbf{u}^l) \\ \times \log \frac{\prod_{i=1}^{N} \mathrm{p}(y_i|f_i^L) \prod_{l=1}^{L} \mathrm{p}(\mathbf{f}^l|\mathbf{u}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1})\mathrm{p}(\mathbf{u}^l; \mathbf{z}^{l-1})}{\prod_{l=1}^{L} \mathrm{p}(\mathbf{f}^l|\mathbf{u}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1})\mathrm{q}(\mathbf{u}^l)} \prod_l \mathrm{d}\mathbf{u}^l \mathrm{d}\mathbf{f}^l \\ = \sum_{i=1}^{n} \mathbb{E}_{\mathrm{q}(f_i^L)}[\log \mathrm{p}(y_i|f_i^L)] - \sum_{l=1}^{L} \mathrm{KL}(\mathrm{q}(\mathbf{u}^l)||\mathrm{p}(\mathbf{u}^l; \mathbf{z}^{l-1})). \qquad (16)\end{aligned}$$

Now the optimization process of the above Evidence Lower Bound (ELBO) becomes even more involved. We have to estimate, together with the model parameters for each layer, the parameters of the distributions $\mathrm{q}(\mathbf{u}^l)$ and the inducing point locations $\mathbf{z}^l$.

The second term is tractable, as the KL divergence between Gaussians is known. However, the expectation involves the marginals of the posterior at the last layer, $\mathrm{q}(f_i^L)$. As we will now see, although this distribution is analytically intractable, it can be sampled efficiently using univariate Gaussians.

Marginalizing out the inducing points in eq. (15), the posterior for the GP layers $\{\mathbf{f}^l\}_{l=1}^L$ is

$$\mathrm{q}(\{\mathbf{f}^l\}_{l=1}^L) = \prod_{l=1}^{L} \mathrm{q}(\mathbf{f}^l|\mathbf{m}^l, \mathbf{S}^l; \mathbf{f}^{l-1}, \mathbf{z}^{l-1}) = \prod_{l=1}^{L} \mathcal{N}(\mathbf{f}^l|\tilde{\boldsymbol{\mu}}^l, \tilde{\boldsymbol{\Sigma}}^l), \qquad (17)$$

where the vector $\tilde{\boldsymbol{\mu}}^l$ is given by $[\tilde{\boldsymbol{\mu}}^l]_i = \mu_{\mathbf{m}^l, \mathbf{z}^{l-1}}(f_i^{l-1})$ and the $n \times n$ matrix $\tilde{\boldsymbol{\Sigma}}^l$ by $[\tilde{\boldsymbol{\Sigma}}^l]_{ij} = \Sigma_{\mathbf{S}^l, \mathbf{z}^{l-1}}(f_i^{l-1}, f_j^{l-1})$. The specific form of the functions $\mu_{\mathbf{m}^l, \mathbf{z}^{l-1}}$ and $\Sigma_{\mathbf{S}^l, \mathbf{z}^{l-1}}$ can be found in [38, Eqs. (7-8)]. Although the distribution in eq. (17) is fully coupled between layers (and thus the posterior in the last layer is analytically intractable), the i-th marginal at each layer $\mathcal{N}(f_i^l|[\tilde{\boldsymbol{\mu}}^l]_i, [\tilde{\boldsymbol{\Sigma}}^l]_{ii})$ only depends on the corresponding i-th input of the previous layer. This allows one to recursively sample $\hat{f}_{i,:}^1 \to \hat{f}_{i,:}^2 \to \cdots \to \hat{f}_{i,:}^L$ from all the layers up to the last one by means of univariate Gaussians. Specifically, $\varepsilon_i^l \sim \mathcal{N}(0,1)$ is first sampled and then for $l = 1, \ldots, L$:

$$\hat{f}_{i,:}^l = \mu_{\mathbf{m}^l, \mathbf{z}^{l-1}}(\hat{f}_{i,:}^{l-1}) + \varepsilon_i^l \cdot \sqrt{\Sigma_{\mathbf{S}^l, \mathbf{z}^{l-1}}(\hat{f}_{i,:}^{l-1}, \hat{f}_{i,:}^{l-1})}. \qquad (18)$$

In summary, the expectation $\mathbb{E}_{\mathrm{q}(f_i^L)}[\log \mathrm{p}(y_i|f_i^L)]$ in the ELBO (see eq. (16)) can be approximated with a Monte Carlo sample generated with eq. (18). Since the ELBO factorizes across data points and the samples can be drawn independently for each point i, scalability is achieved through sub-sampling the data in mini-batches. The complexity to evaluate the ELBO and its gradients is $\mathcal{O}(nm^2L)$. The code is integrated within GPflow (a GP framework built on top of Tensorflow) and is publicly available[3].

[3] https://github.com/ICL-SML/Doubly-Stochastic-DGP

To predict in a new x_*, eq. (18) is used to sample S times[4] from the posterior up to the $(L-1)$-th layer using the test location as initial input. This yields a set $\{f_*^{L-1}(s)\}_{s=1}^S$ with S samples. Then, the density over f_*^L is given by the Gaussian mixture (recall that all the terms in eq. (17) are Gaussians):

$$\mathrm{q}(f_*^L) = \frac{1}{S}\sum_{s=1}^{S} \mathrm{q}(f_*^L|\mathbf{m}^L, \mathbf{S}^L; f_*^{L-1}(s), \mathbf{z}^{L-1}).$$

6. Experiments

In this section we carry out an exhaustive evaluation of the proposed classification approach which, as we have already indicated, is based on the use of GPs and DGPs and granulometry profiles on OD H&E images. First, we compare the classification performance of GPs with the most popular shallow classifiers using classical texture descriptors, granulometry profiles and a combination of them extracted from OD H&E images. To show the importance of the space where images are represented, we replicate the experiments using RGB H&E images. Once we show that features should be extracted from OD H&E images and that our approach is the best performing one when only shallow classifiers are used, we proceed to compare it to state-of-art deep learning strategies based on a variety of pre-trained CNNs. To demonstrate the generalization capability of the patch-wise trained model, we carry out a validation at WSI level (for the test set). We predict the per pixel probability of being cancerous and validate the obtained probability map.Despite being much simpler, GPs and DGPs perform similarly and they are also competitive to the tested deep classifiers. In other words, the quality of our OD extracted features does not require more than a single layer GP toobtain excellent results. Finally, an external validation has been carried out to assess the competitiveness of the proposed descriptor together with the GP classifier against other models.

6.1. Feature extraction

As feature descriptors we computed the morphological descriptors PS_Φ and PS_Γ on H and E, respectively, and their geodesic versions PS_Φ^r and PS_Γ^r. PS_Φ and PS_Φ^r with SE of increasing size in steps of $s=2$ from 0 to $n_{max}=24$, and in steps of $s=4$ for PS_Γ and PS_Γ^r from 0 to $n_{max}=48$. Note that we use *Gran* and *GeoGran* labels to denote PS and PS^r descriptors, respectively. Besides that, to capture the texture information we use the uniform and rotationally invariant Local Binary Patterns (LBP) [41] as baseline descriptor (with neighbourhood of $R=1$ and $P=8$) and the combination of it with a contrast measure, according to the work of Guo et al. [42], obtaining an additional Local Binary Pattern Variance ($LBPV$) descriptor. The different combinations of descriptors have been labelled as *GranLBP*, *GranLBPV*, *GeoGranLBP* and *GeoGranLBPV*.

[4] Results become stable after a few samples. Here, S was set to 100.

Table 2: Performance of descriptors and classifiers in RGB space with a 512^2 patch size.

AUC	RF	GP	XgBoost
LBP	0.6663 ± 0.1400	0.7003 ± 0.1190	0.6728 ± 0.1279
LBPV	0.7695 ± 0.0565	0.8243 ± 0.0891	0.7912 ± 0.0674
Gran	0.8549 ± 0.0856	0.8984 ± 0.0641	0.8778 ± 0.0735
GeoGran	0.9089 ± 0.0494	0.8910 ± 0.0599	0.9095 ± 0.0454
GranLBP	0.8331 ± 0.0949	0.9111 ± 0.0492	0.8551 ± 0.0842
GranLBPV	0.8758 ± 0.0611	0.9280 ± 0.0349	0.8908 ± 0.0509
GeoGranLBP	0.8958 ± 0.0566	0.9014 ± 0.0507	0.9048 ± 0.0469
GeoGranLBPV	0.9174 ± 0.0351	$\mathbf{0.9307 \pm 0.0307}$	0.9273 ± 0.0329

Table 3: Performance of descriptors and classifiers in OD space with a 512^2 patch size.

AUC	RF	GP	XgBoost
LBP	0.9300 ± 0.0603	0.9253 ± 0.0635	0.9262 ± 0.0615
LBPV	0.9351 ± 0.0373	0.9443 ± 0.0314	0.9421 ± 0.0243
Gran	0.9323 ± 0.0453	0.9516 ± 0.0346	0.9461 ± 0.0322
GeoGran	0.9690 ± 0.0303	0.9636 ± 0.0242	0.9688 ± 0.0249
GranLBP	0.9436 ± 0.0640	0.9581 ± 0.0422	0.9541 ± 0.0524
GranLBPV	0.9370 ± 0.0340	0.9696 ± 0.0175	0.9573 ± 0.0206
GeoGranLBP	0.9666 ± 0.0408	0.9669 ± 0.0283	0.9700 ± 0.0304
GeoGranLBPV	0.9692 ± 0.0241	$\mathbf{0.9807 \pm 0.0097}$	0.9747 ± 0.0170

6.2. Comparison of shallow classifiers

To demonstrate the superiority of nonparametric probabilistic models based on GPs and morphological features we compare GPs with different state-of-art shallow classifiers on different extracted features. We compare the performance of the models on OD and RGB spaces, testing two patch sizes, 512^2 and 1024^2.

We use variational inference on a single-layer GP classifier with a RBF kernel. We utilize a sparse model with 800 inducing points when the patch size is 512^2. For 1024^2 patch size we do not utilize inducing points. For comparison, we use Random Forest (RF) and Extreme Gradient Boosting (XgBoost). These tree-based ensemble models can capture complex patterns in data. They are state-of-art shallow classifiers.

For each classifier we applied a five-fold cross-validation to validate and compare the performance of the proposed granulometry descriptors (using the described classifiers). Patches coming from the same image and the same patient were assigned to the same fold. Consequently, we avoided correlation between training and test sets which would distort the results. Due to the nature of prostatic images, the amount of benign instances is significantly greater than the cancerous ones. To deal with this imbalanced scenario, we built several classifiers with the positive instances and a subset of the negative ones so that

Table 4: Performance of descriptors and classifiers in RGB space with a 1024^2 patch size.

AUC	RF	GP	XgBoost
LBP	0.6279 ± 0.1751	0.6900 ± 0.1841	0.6460 ± 0.1660
LBPV	0.7517 ± 0.0847	0.8222 ± 0.1169	0.7638 ± 0.0934
Gran	0.8018 ± 0.1166	0.8785 ± 0.0525	0.8177 ± 0.1071
GeoGran	0.9269 ± 0.049	0.9242 ± 0.0398	0.9242 ± 0.0425
GranLBP	0.7910 ± 0.1379	0.8780 ± 0.0512	0.7955 ± 0.1437
GranLBPV	0.8471 ± 0.0820	$\mathbf{0.9447 \pm 0.0252}$	0.8536 ± 0.0708
GeoGranLBP	0.9079 ± 0.0675	0.9062 ± 0.0462	0.9146 ± 0.0478
GeoGranLBPV	0.9338 ± 0.0339	0.9293 ± 0.0510	0.9289 ± 0.0347

Table 5: Performance of descriptors and classifiers in OD space with a 1024^2 patch size.

AUC	RF	GP	XgBoost
LBP	0.9433 ± 0.0615	0.9353 ± 0.0661	0.9350 ± 0.0640
LBPV	0.9244 ± 0.0671	0.9684 ± 0.0217	0.9419 ± 0.0575
Gran	0.9408 ± 0.0493	0.9635 ± 0.0320	0.9590 ± 0.0448
GeoGran	0.9826 ± 0.0237	0.9824 ± 0.0165	0.9814 ± 0.0256
GranLBP	0.9525 ± 0.0654	0.9647 ± 0.0488	0.9578 ± 0.0603
GranLBPV	0.9318 ± 0.0480	0.9736 ± 0.0211	0.9553 ± 0.0386
GeoGranLBP	0.9760 ± 0.0366	0.9800 ± 0.0230	0.9800 ± 0.0277
GeoGranLBPV	0.9789 ± 0.0187	$\mathbf{0.9855 \pm 0.0089}$	0.9764 ± 0.0218

each classifier faces a balanced problem being the final prediction the average of the predictions of each classifier. The evaluation metric we selected to compare the performance of different methods is the area under the ROC curve (AUC).

Tables 2, 3 (512^2) and 4, 5 (1024^2) summarize the obtained results. Analysing all the tables, we observe that, in both spaces, key tumoral information is better encoded by morphological than by texture features. More in depth, *LBPV* and *GeoGran* perform better than *LBP* and *Gran* in both spaces. Regarding the classifiers, GPs discriminate better than the others for all patch sizes and spaces.

For every descriptor and classifier, the results obtained in the OD space are superior to those achieved in the RGB space. This is the space used by the majority of current state-of-the-art methods. Moreover, texture and morphological information for classification purposes are better captured in the OD space.

In summary, for both patch sizes, the best results are obtained in the OD space when *GeoGranLBPV* are the input to a GP classifier. The obtained AUCs are 0.9807 (512^2) and 0.9855 (1024^2). This fact suggests that texture and morphology features provide complementary information to characterize prostatic tumoral tissues. In the coming section we compare GPs and DGPs, using the best performing features, to CNNs.

6.3. Comparison of deep classifiers

The previous experiment indicates that the proposed geodesic granulometries (*GeoGran*) in combination with texture information (*LBPV*) allows us to create a descriptor *GeoGranLBPV* able to accurately classify histopathological tissues using GPs. We now compare GPs and DGPs used on *GeoGranLBPV* extracted from OD images to CNNs used on raw images. Three of the most well-known deep convolutional neural networks for image classification: VGG19 [43], Xception [44] and Inception v3 [45] are utilized. The main reason to select these CNNs was their wide use in the detection of tumoral tissues in histological images [46, 10, 11, 12, 13, 14].

For this comparison, the cross validation setup used for shallow classifiers was utilized. Together with the two GPs described in the previous section, a three-layer DGP classifier [38] with RBF kernel was used on the extracted features. Our model employs 100 inducing points per layer. Although with shallow GPs we achieved a very good performance, the DGP is used here as a nonparametric multi-layer classification model to carry out a comparison between the deep structure of VGG19, Xception, and Inception v3 and a GP based counterpart.

The parameters of the CNN were optimized following the procedures described in Table 6. In this experiment, due to the reduced number of samples of our data set, we fine-tuned the architectures, initializing them with the best weights obtained in the ImageNet challenge [47] and re-trained them using our raw RGB histological images as input. The re-training process was performed using the binary cross entropy loss function, from the layers indicated in Table 6 to the end of the networks. Early stopping, with fifteen epochs of patience value, was used to prevent overfitting. Synthetic data was automatically created

Table 6: Empirically-tuned hyperparameters for Inception v3, Xception and VGG19.

Architecture	**Layer name**	**Optimizer**	**Learning rate**
VGG19	'block3_conv1'	Stochastic Gradient Descent	$1 \cdot 10^{-4}$
Inception v3	'mixed7'	Nesterov Adam	$1 \cdot 10^{-5}$
Xception	'add10'	Stochastic Gradient Descent	$1 \cdot 10^{-4}$

using data augmentation methods (i.e. rotating, flipping, rescaling, translating, etc.) and a batch size of 16 samples, constrained by the available memory of the NVIDIA Titan V GPU utilized in this work, was used.

Table 7: Performance of Deep Classifiers for 512^2 patch size.

	Inception v3	VGG19	Xception	DGP
AUC	0.9196 ± 0.0302	0.9813 ± 0.0068	0.921 ± 0.026	**0.9829 ± 0.0092**

The average metric values for the five-fold comparison of deep models are reported in Tables 7 (512^2 patch size) and 8 (1024^2 patch size). As it can be observed from these tables, the morphological and textural information encoded by our proposed hand-crafted descriptor compares well to the automatic features directly learned by the CNNs from the data.

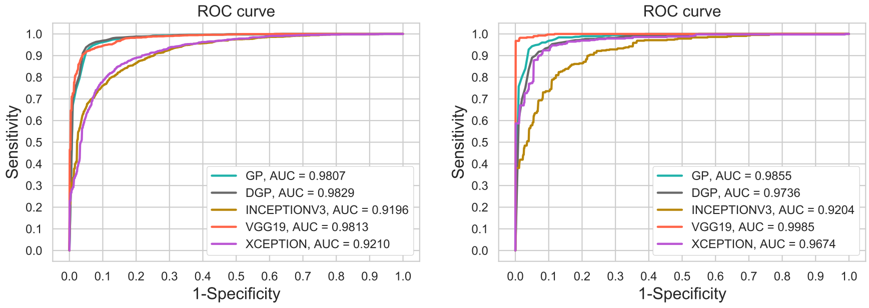
For 512^2 patch size (see Table 7), the hand-driven learning by DGP outperforms Inception v3 and Xception models in terms of AUC values by 6.33% and 6.19%, respectively. Additionally, the proposed methodology performs similarly to VGG19. The obtained AUC is 0.9829 which is slightly better than the one obtained by the shallow GP (0.9807), this suggests that our hand-crafted features are good enough to perform an excellent classification and they do not require more than the use of a well grounded nonparametric single layer classifier with no parameter tuning. Figure 6a shows the ROC curves corresponding to these deep classifiers together with the single layer GP used in the previous section for the 512^2 case.

When the patch size is 1024^2, see Table 8, VGG19 outperforms the rest of the deep classifiers. Its corresponding AUC is 0.9985 which is slightly better than the ones obtained by our DGP (0.9736) and GP (0.9855). Figure 6b shows the ROC curves corresponding to these deep classifiers together with the one-layer GP used in the previous section for the 1024^2 case. Note again our approach does not seem to need more than a layer to obtain excellent results.

Regarding computational cost, the proposed methodology needs less time

Table 8: Performance of Deep Classifiers for 1024^2 patch size.

	Inception v3	VGG19	Xception	DGP
AUC	0.9204 ± 0.0525	**0.9985 ± 0.0009**	0.9674 ± 0.0194	0.9736 ± 0.0239



(a) (b)

Figure 6: ROC curve plot for all deep classifiers together with the best performing shallow one: for (a) 512^2 and (b) 1024^2 patch size.

Table 9: Analysis of the patch-wise (512^2) computational cost for the deep models and shallow GPs. Time measured for Deep GPs and GPs includes the feature extraction and classification steps. Note that CNNs were trained and tested in a Titan V GPU while these tasks were performed in the CPU for GPs and DGPs.

Time (sec.)	**VGG19**	**Inception v3**	**Xception**	**Deep GPs**	**GPs**
Training	28742.71	24321.12	23441.33	14587.1362 + 4431.1845 = 19018.3207	14587.1362 + 550.2587 = 15137.3949
Inference	0.8522	0.7873	0.7177	1.5753 + 0.0357 = 1.611	1.5753 + 0.0003 = 1.581

than the deep learning-based approaches in the training stage (see Table 9). It is important to remark that CNN models require specific hardware to be trained in an affordable time interval while GPs and DGPs just need a CPU to be trained. Due to this fact the inference phase in a CNN model requires less time than the proposed hand-driven approach. The computational time analysis was performed on an Intel i7@3.10 GHz of 16 GB of RAM with an NVIDIA GeForce Titan V to train VGG19, Inception v3, and Xception CNNs. Python 3.5 was the language used and the libraries GPflow and Keras were used for GPs and DGPs and deep learning methods, respectively.

6.4. Whole Slide Image evaluation

Our ultimate goal is to provide pathologists with useful tools for WSI analysis. With this aim, we extend the patch-wise classification model to WSI classification, trying to identify cancerous areas in unseen WSIs. Following the approach in [8], we split each biopsy of the WSIs into overlapping patches. For each pixel, we estimate the probability of being cancerous by bilinearly interpolating the predicted probabilities of the four closest patches (in terms of euclidean distance to the center of the patches). With this pixel-wise classification, we obtain a probability map per each biopsy of a WSI (see Figure 8(b)). To assess the generalization capability of our model we used the 19 WSIs in the test set: 17 malignant and 2 benign. The magnification factor was, like during training, 10×. The overlap between patches was 75%, for both, 512^2 and 1024^2, patch sizes. We compare GP and DGP + *GeoGranLBPV* extracted in the OD

Table 10: Sensitivity for 1, 2 and 3 false positives for 512^2 and 1024^2 patch sizes.

Sensitivity	512^2			1024^2		
	1 FP	2 FP	3 FP	1 FP	2 FP	3 FP
GP	0.8387	0.9489	**1**	**0.5606**	**0.9277**	0.9804
DGP	0.8340	0.9492	**1**	0.4710	0.8993	**0.9920**
Inception v3	0.6985	0.9125	0.9519	0.4763	0.7981	0.9715
Xception	0.8081	0.9589	0.9984	0.5342	0.8115	0.9248
VGG19	**0.8610**	**0.9972**	**1**	0.5084	0.8089	0.9171

space to the models obtained by fine-tuning the three CNNs. All patch-wise models were trained using the 60 images in the training set. For WSI based evaluation, the free-response receiver operating characteristic (FROC) curve, defined as sensitivity versus the average number of false-positives per image, was used. After CAMELYON16 challenge [5], FROC is widely used for image level cancer detection evaluation.

Table 10 shows, for both patch sizes, the sensitivity of each model for 1, 2 and 3 false cancerous regions. The results have been averaged over the 17 malignant testing WSIs: these WSIs contain both benign and malignant glands in addition to different cancer grades. These images present a high inflammation so it is a challenging task to detect well the benign glands. All models (CNN-based together with GP and DGP) generalize worse for 1024^2 patch size. This is probably due to the reduced sample size which may lead to overfitting during training and poor generalization during testing. Notice, however, that for this reason, the probabilistic and nonparametric nature of our GP and DGP models leads to a better generalization capability for this size. For a 512^2 patch size, we see that VGG19 performs slightly better than GP and DGP while Xception is a bit worse. Inception v3 generalizes poorly compared to the rest. Indeed, VGG19, GP and DGP are the only methods that detect all cancer pixels with a cost of 3 false positives areas for each pixel correctly classified. Figure 7 depicts the FROC for all compared models (512^2 and 1024^2 patch size) and clearly shows that our approach is competitive to state-of-art CNN architectures.

In Figure 8, for 512^2 patch size, we can compare the probability maps obtained by the best performing model (GP) (Figure 8(a)), and the cancerous regions annotated by the pathologist (Figure 8(b)). The probability maps are represented as heat maps, where red and blue colors indicate the highest and the lowest probabilities of being cancerous, respectively. The zoomed in regions show that the highest probabilities (redish colors) obtained by our model are in agreement with the cancerous areas marked by the experts while at the boundary the probability decreases. Besides, the proposed model can discriminate successfully whether a gland is benign or malignant in the same WSI giving zero or low probability to benign glands. For a more complete study, in Figure

[5] https://camelyon16.grand-challenge.org/Home/

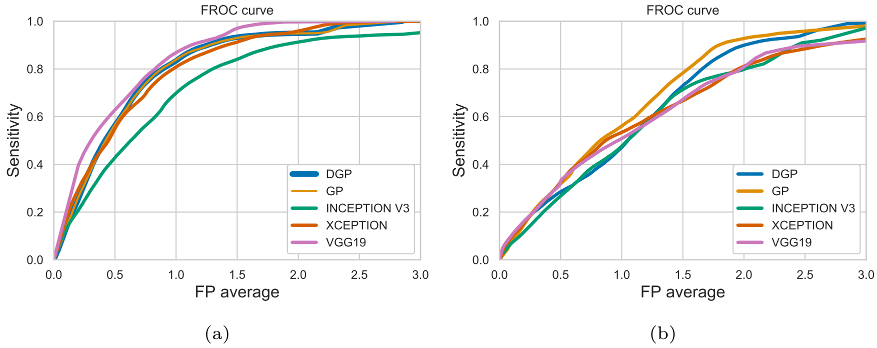
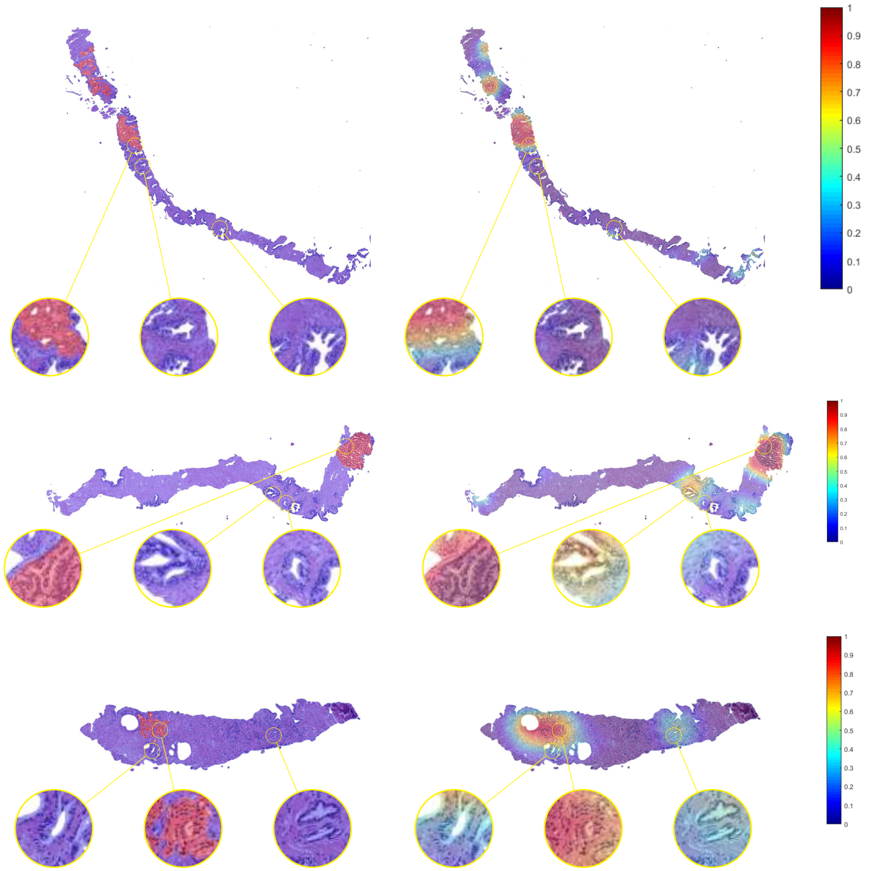


Figure 7: FROC for CNN-based and GP and DGP models: (a) 512^2 and (b) 1024^2 patch size.

9, we show the prediction of the proposed GP model in 3 regions of the two benign samples in the test subset. Since the heat maps give to each image a very low probability of being cancerous, this model does not suffer from false positives in benign WSIs.

Regarding computational cost and model complexity, taking into account the patch-wise average time (see Table 9) and the average number of patches resulting from all the biopsies contained in the testing WSIs (see Section 2), we can calculate the average time to predict a new WSI. Xception is the fastest model in obtaining the probability map for a WSI, in particular, the expected time ranges from 4.3 to 5.7 minutes depending on whether the WSI is composed of three or four biopsies. The Xception fine-tuning process is performed on 8,406,458 trainable parameters and the storage space of the model is 147.6 MB. Inception v3 model has 12,816,002 trainable parameters and the storage space of the model is 186.2 MB. The inference time ranges from 4.7 to 6.24 minutes. VGG19 takes around 5.1 to 6.8 minutes for WSIs with three and four biopsies, respectively. The fine-tuning process is performed on 130,923,522 trainable parameters and the storage space of the model is 1.02 GB. The models with the highest ability of generalization, i.e. models based on gaussian processes, spend around 9.3 and 12.7 minutes to compute the resulting probability map for a WSI composed of three and four biopsies, respectively. The number of GP and DGP parameters is 2,672,008 and 339,644 (due to the use of a less number of inducing points for DGP), respectively. The storage space is 20.88 MB for GP model and 10.10 MB for DGP model. As we have already indicated, notice that DL-based methods are computed in a Titan V GPU while our hand-driven learning approaches are run in a i7 core.

Analysing the obtained computational cost, the model complexity and the performance of the models on new samples (see Table 10), we conclude that the proposed approaches based on GPs reach an interesting trade-off between these three capabilities. It is important to highlight that the task of diagnosing biopsies is an offline process and spending six additional minutes (additional DGP computational time in comparison to Inception v3 for a WSI with four



(a) (b)

Figure 8: GP model validation in slides with cancer: (a) Cancerous areas annotated by the pathologists (ground truth); (b) Probability maps (heat maps) obtained by the proposed GP model.

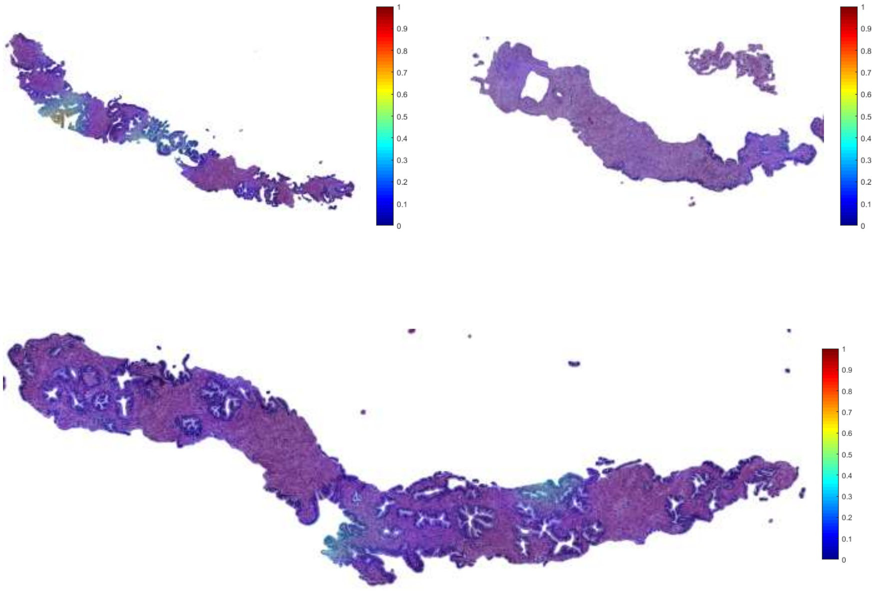


Figure 9: GP model validation in slides with benign glands: Probability maps (heat maps) obtained by the proposed GP model

biopsies) pays off due to the increased sensitivity. See in Table 10 the 24% improvement for 1FP for 512^2 patch size. In addition, the GP based models are, with regard to number of parameters and space, four (GPs) and five (DGP) times (DGPs) less expensive than the best CNN-based approach (VGG19).

6.5. Validation on an external data

To analyze and corroborate the robustness and generalization power of the proposed methodology, we also evaluate all the models on an external database. We have used the prostate cancer database proposed by Gertych et al. [15, 29]. This database includes 625 patches with different grades and combinations of them. No spatial information of these patches in the WSI is provided. The size of the patches at 20× magnification is 1201^2. Each patch has a mask with annotation provided by pathologists (see Figure 10). This mask indicates the class of each pixel: stroma, benign or malignant (distinguishing between grade 3, 4, and 5).

The GP model was trained using the SICAPv1 database and tested on the Gertych et al. [15, 29] database. Since we use for training 512^2 patches at 10× magnification, we downsampled the test patches to a 10× magnification and cropped the central region of 512^2 size. We labelled each patch of the test set as benign if there are no malignant pixels in the image. Patches with more than 20% malignant pixels (this information is provided by the mask) are classified

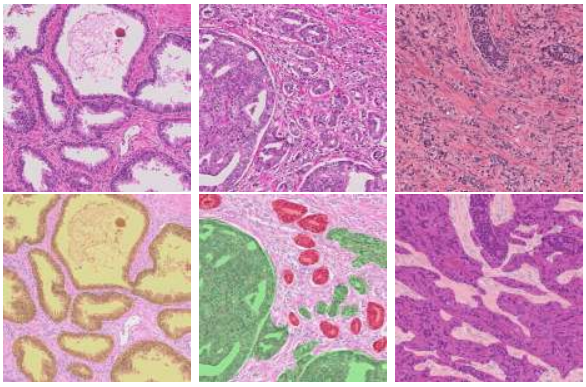


Figure 10: Patches from the external database [15, 29]. The colored masks indicate the annotated classes by the pathologist in this database: white (stroma), yellow (benign), red (grade 3), green (grade 4) and purple (grade 5).

Table 11: Performance of descriptors and classifiers in the OD space on the external database

AUC	RF	GP	XgBoost	DGP
LBP	0.8490	0.7529	0.7464	0.7833
LBPV	0.8415	0.6869	0.8593	0.6867
Gran	0.8572	0.8775	0.8851	0.8156
GeoGran	0.8828	**0.9249**	0.8636	0.8471
GranLBP	0.8629	0.8624	0.8643	0.6913
GranLBPV	0.8494	0.7998	0.8811	0.8850
GeoGranLBP	0.8757	0.8766	0.8754	0.8221
GeoGranLBPV	0.8872	0.7645	0.8365	0.8010

as malignant (for the binary classification approach proposed). This results in 593 patches of which 244 are benign and 349 are pathological.

The obtained results are reported in Tables 11 and 12 for the OD and RGB spaces, respectively. The morphological features (*Gran* and *GeoGran*) outperform those based on texture (*LBP* and *LBPV*) in both RGB and OD spaces independently of the chosen classifier. Furthermore, in almost all cases, the OD space outperforms the RGB space. In this experiment combining texture and morphological descriptors does not achieve better results except in a few cases, for example, *GeoGranLBPV* + DGP in RGB space which obtains the best result in this space. However, the proposed descriptor based on geodesic granulometry *GeoGran* using GP as the classifier in the OD space outperforms the rest with an AUC of 0.9249.

These results indicate the robustness and generalization capabilities of the proposed morphological descriptor on different datasets. They also indicate that texture based features perform worse. This may have been exacerbated by the fact that white balancing was not performed on the second dataset since only

Table 12: Performance of descriptors and classifiers in RGB space on the external database

AUC	RF	GP	XgBoost	DGP
LBP	0.3444	0.3336	0.7051	0.2840
LBPV	0.6122	0.3116	0.7285	0.6597
Gran	0.7251	0.6473	0.7367	0.5928
GeoGran	0.8674	0.7130	0.8507	0.8026
GranLBP	0.5536	0.1214	0.7292	0.2728
GranLBPV	0.6346	0.3048	0.6622	0.8310
GeoGranLBP	0.8597	0.2756	0.8101	0.8158
GeoGranLBPV	0.8746	0.8097	0.8392	**0.8902**

Table 13: Performance of Deep Classifiers on the external database.

	Inception v3	VGG19	Xception
AUC	0.8846	**0.9714**	0.8670

patches were provided. We also verified that the OD space is more informative than the RGB one for most of the descriptors/classifiers used in the four studies carried out in this work. Furthermore, the GP is the classifier which shows the best performance.

Finally, for a complete comparison, the performance of deep neural networks in this database is reported in Table 13. We can see that VGG19 obtains the best results. Notice, however, that the size of this model exceeds the Gigabyte in contrast to GP models which can be stored in much smaller disks (21 MB). Notice also that VGG19 is a well established architecture while the best DGPs is still work in progress. Regarding the other architectures (i.e. Inception v3 and Xception), our proposed descriptor *GeoGran* performs better using the probabilistic classifier based on a single-layer GP on the OD space, improving by a 4% and 6%, respectively. This demonstrates the competitive ability to capture cancer patterns with respect to state-of-art CNNs, even in databases that have never been seen by the classifier.

7. Conclusions and future work

In this work, we have proposed a novel descriptor to characterize and differentiate benign and pathological regions in histological prostate images. This descriptor registers the granularity of the tissue elements without previous segmentation.

We have shown that features should be extracted from OD H&E images, where our OD geodesic granulometry descriptor reveals the importance of the stroma identifying cancer. We have also shown that GP is the best performing classifier when only shallow classifiers are used. The best performing features (*GeoGranLBPV*) and the best performing shallow classifier (GP) together with its multilayer version (DGP) have then been compared to state-of-art deep learning strategies based on a variety of pre-trained CNNs. To analyze the generalization capability of the patch-wise trained model, we have carried out a validation at WSI level. We have predicted the per pixel probability of being cancerous and validate the obtained probability map. GPs and DGPs perform similarly and, furthermore, they are also competitive to the tested deep classifiers identifying successfully cancer in WSIs. To assess the robustness and generalization capabilities of the proposed descriptor, an external database has been utilized. The obtained results corroborate the quality of the proposed descriptor when combined with a GP based classifier. In summary, we have shown that our OD extracted features do not require more than a single layer GP to outperform the best performing shallow classifiers and to be competitive to deep classifiers.

Additionally, we have created a public database (SICAPv1) that includes original WSIs and labels annotated by expert pathologists.

As future work, the use of geodesic granulometries and multi-class DGP for the automatic detection of Gleason grade in histopathological images will be addressed. Moreover, new annotated images will be added to SICAPv1.

References

[1] W. H. Organization, Global cancer observatory (2018).
URL http://gco.iarc.fr/

[2] D. F. Gleason, Histologic grading of prostate cancer: A perspective, Human Pathology 23 (3) (1992) 273 – 279, the Pathobiology of Prostate Cancer-Part 1.

[3] D. Komura, S. Ishikawa, Machine learning methods for histopathological image analysis, Computational and Structural Biotechnology Journal 16 (2018) 34 – 42.

[4] S. Wang, K. Burtt, B. Turkbey, P. L. Choyke, R. M. Summers, Computer aided-diagnosis of prostate cancer on multiparametric mri: A technical review of current research, in: BioMed research international, 2014.

[5] S. Roy, A. K. Jain, et al., A study about color normalization methods for histopathology images, Micron 114 (2018) 42–61.

[6] A. C. Ruifrok, D. A. Johnston, Quantification of histochemical staining by color deconvolution, Analytical and quantitative cytology and histology 23 (4) (2001) 291—299.

[7] V. Gupta, A. Bhavsar, Breast cancer histopathological image classification: Is magnification important?, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 769–776.

[8] G. Litjens, C. I. Sánchez, N. Timofeeva, M. Hermsen, I. Nagtegaal, I. Kovacs, C. Hulsbergen van de Kaa, P. Bult, B. van Ginneken, J. van der Laak, Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis 6 (2016) 26286.

[9] L. Hou, D. Samaras, T. M. Kurç, Y. Gao, J. E. Davis, J. H. Saltz, Patch-based convolutional neural network for whole slide tissue image classification, in: Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, 2016, pp. 2424–2433.

[10] S. Kwok, Multiclass classification of breast cancer in whole-slide images, in: Image Analysis and Recognition, Springer International Publishing, Cham, 2018, pp. 931–940.

[11] I. Koné, L. Boulmane, Hierarchical resnext models for breast cancer histology image classification, in: Image Analysis and Recognition, Springer International Publishing, Cham, 2018, pp. 796–803.

[12] B. Marami, M. Prastawa, M. Chan, M. Donovan, G. Fernandez, J. Zeineh, Ensemble network for region identification in breast histopathology slides, in: Image Analysis and Recognition, Springer International Publishing, Cham, 2018, pp. 861–868.

[13] M. Ferlaino, C. A. Glastonbury, C. Motta-Mejia, M. Vatish, I. Granne, S. Kennedy, C. M. Lindgren, C. Nellåker, Towards deep cellular phenotyping in placental histology, CoRR abs/1804.03270 (2018).

[14] Shallu, R. Mehra, Breast cancer histology images classification: Training from scratch or transfer learning?, ICT Express 4 (4) (2018) 247 – 254.

[15] N. Ing, Z. Ma, J. Li, H. Salemi, C. W. Arnold, B. S. Knudsen, A. Gertych, Semantic segmentation for prostate cancer grading by convolutional neural networks, in: SPIE Medical Imaging, Vol. 10581, 2018.

[16] W. Li, J. Li, K. V. Sarma, K. C. Ho, S. Shen, B. S. Knudsen, A. Gertych, C. W. Arnold, Path r-cnn for prostate cancer diagnosis and gleason grading of histological images, IEEE Transactions on Medical Imaging 38 (2018) 945–954.

[17] R. Srivastava, R. Kumar, S. Srivastava, Detection and classification of cancer from microscopic biopsy images using clinically significant and biologically interpretable features, Journal of Medical Engineering (08 2015).

[18] O. Oğuz, A. E. Çetin, R. c. Atalay, Classification of hematoxylin and eosin images using local binary patterns and 1-d sift algorithm, in: IWCIM, Vol. 2, 2018.

[19] L. Gorelick, O. Veksler, M. Gaed, J. A. Gómez, M. Moussa, G. Bauman, A. Fenster, A. D. Ward, Prostate histopathology: Learning tissue component histograms for cancer detection and classification, IEEE Transactions on Medical Imaging (TMI) 32 (10) (2013) 1804–1818.

[20] M. T. Farooq, A. Shaukat, U. Akram, O. Waqas, M. Ahmad, Automatic gleason grading of prostate cancer using gabor filter and local binary patterns, in: 2017 40th International Conference on Telecommunications and Signal Processing (TSP), 2017, pp. 642–645.

[21] M. Dinesh Kumar, M. Babaie, S. Zhu, S. Kalra, H. R. Tizhoosh, A comparative study of cnn, bovw and lbp for classification of histopathological images, in: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), 2017, pp. 1–7.

[22] K. Nguyen, A. Sarkar, A. K. Jain, Prostate cancer grading: Use of graph cut and spatial arrangement of nuclei, IEEE Transactions on Medical Imaging (TMI) 33 (07 2014).

[23] S. Doyle, M. Hwang, K. Shah, A. Madabhushi, M. Feldman, J. Tomaszeweski, Automated grading of prostate cancer using architectural and textural image features, in: 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2007, pp. 1284–1287.

[24] J. T. Kwak, S. M. Hewitt, Multiview boosting digital pathology analysis of prostate cancer, Computer Methods and Programs in Biomedicine 142 (2017) 91–99.

[25] J. Ren, E. Sadimin, D. J. Foran, X. Qi, Computer aided analysis of prostate histopathology images to support a refined gleason grading system, in: Proceedings Volume 10133, Medical Imaging 2017: Image Processing, 2017.

[26] N. Zhou, A. Fedorov, F. Fennessy, R. Kikinis, Y. Gao, Large scale digital prostate pathology image analysis combining feature extraction and deep neural network, arXiv e-prints (2017) arXiv:1705.02678.

[27] M. Valkonen, K. Kartasalo, K. Liimatainen, M. Nykter, L. Latonen, P. Ruusuvuori, Metastasis detection from whole slide images using local features and random forests, Cytometry. Part A : the journal of the International Society for Analytical Cytology 91 (04 2017).

[28] A. Pimkin, G. Makarchuk, V. Kondratenko, M. Pisov, E. Krivov, M. Belyaev, Ensembling neural networks for digital pathology images classification and segmentation, in: Image Analysis and Recognition - 15th International Conference, ICIAR 2018, 2018, pp. 877–886.

[29] A. Gertych, N. Ing, Z. Ma, T. J. Fuchs, S. Salman, S. Mohanty, S. Bhele, A. Velásquez-Vacca, M. B. Amin, B. S. Knudsen, Machine learning approaches to analyze histological images of tissues from radical prostatectomies, Computerized Medical Imaging and Graphics 46 (2015) 197 – 208, information Technologies in Biomedicine.

[30] K. Rajpoot, N. Rajpoot, Svm optimization for hyperspectral colon tissue cell classification, in: Medical Image Computing and Computer-Assisted

Intervention (MICCAI) 2004, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 829–837.

[31] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2006.

[32] M. Kandemir, C. Zhang, F. A. Hamprecht, Empowering multiple instance histopathology cancer diagnosis by cell graphs, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014, Springer International Publishing, Cham, 2014, pp. 228–235.

[33] A. Damianou, N. Lawrence, Deep Gaussian processes, in: Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, Vol. 31 of Proceedings of Machine Learning Research, PMLR, Scottsdale, Arizona, USA, 2013, pp. 207–215.

[34] M. Kandemir, Asymmetric transfer learning with deep gaussian processes, in: Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, PMLR, Lille, France, 2015, pp. 730–738.

[35] G. Campanella, V. Werneck Krauss Silva, T. J. Fuchs, Terabyte-scale Deep Multiple Instance Learning for Classification and Localization in Pathology, arXiv e-prints (2018) arXiv:1805.06983.

[36] E. Arvaniti, K. S. Fricker, M. Moret, N. J. Rupp, T. Hermanns, C. Fankhauser, N. Wey, P. J. Wild, J. H. Rueschoff, M. Claassen, Automated gleason grading of prostate cancer tissue microarrays via deep learning, Scientific Reports (2018).

[37] Openseadragon, http://openseadragon.github.io/, accessed: 10-07-2018.

[38] H. Salimbeni, M. Deisenroth, Doubly stochastic variational inference for deep gaussian processes, in: Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 4588–4599.

[39] M. Bauer, M. van der Wilk, C. Rasmussen, Understanding probabilistic sparse Gaussian process approximations, in: Advances in Neural Information Processing Systems, 2016, pp. 1533–1541.

[40] M. Titsias, Variational learning of inducing variables in sparse gaussian processes, in: Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, Vol. 5 of Proceedings of Machine Learning Research, PMLR, 2009, pp. 567–574.

[41] M. Pietikäinen, A. Hadid, G. Zhao, T. Ahonen, Computer Vision Using Local Binary Patterns, Springer, 2011.

[42] Z. Guo, L. Zhang, D. Zhang, Rotation invariant texture classification using lbp variance (lbpv) with global matching, Pattern Recognition 43 (3) (2010) 706 – 719.

[43] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations (ICLR), 2015.

[44] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800–1807.

[45] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.

[46] N. Coudray, P. Santiago Ocampo, T. Sakellaropoulos, N. Narula, M. Snuderl, D. Fenyö, A. L. Moreira, N. Razavian, A. Tsirigos, Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning, Nature Medicine 24 (09 2018).

[47] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision (IJCV) 115 (3) (2015) 211–252.

Chapter 4

Crowdsourcing in histopathological images using Gaussian Processes

4.1 Publication details

Authors: Miguel López-Pérez, Mohamed Amgad, Pablo Morales-Álvarez, Pablo Ruiz, Lee A. D. Cooper, Rafael Molina, and Aggelos K. Katsaggelos.
Title: Learning from Crowds in Digital Pathology using Scalable Variational Gaussian Processes.
Publication: Scientific Reports, vol. 11, no. 1, 1-9, 2021.
Status: Published.
Quality indices:

- Impact Factor (JCR 2020): 4.380.
- Rank: 17/72 (Q1) in Multidisciplinary Sciences.

4.2 Main contributions

- We introduce Gaussian Processes for crowdsourcing classification in digital pathology. We apply crowdsourced GPs to triple-negative breast cancer images.
- Our fully-probabilistic approach estimates the class-conditional expertise of each annotator. We model the behavior of each participant and predict the non-expert annotations.
- This work provides an efficient approach to the labeling of histopathological images.

Learning from crowds in digital pathology using scalable variational Gaussian processes

Miguel López-Pérez[a], Mohamed Amgad[b], Pablo Morales-Álvarez[c], Pablo Ruiz[d], Lee A. D. Cooper[b,e,f,*], Rafael Molina[a], Aggelos K. Katsaggelos[e,f]

[a] *Department of Computer Science and Artificial Intelligence, University of Granada, 18010 Granada, Spain.*

[b] *Department of Pathology at Northwestern University, Chicago, IL, 60611 US.*

[c] *Microsoft Research, Cambridge, CB12FB, UK.*

[d] *OriGen.AI, Brooklyn, NY, 11201 US.*

[e] *Department of Electrical and Computer Engineering at Nothwestern University, Evanston, IL, 60208 US.*

[f] *Center for Computational Imaging and Signal Analytics, Northwestern University, Chicago, IL 60611.*

Abstract

The volume of labeled data is often the primary determinant of success in developing machine learning algorithms. This has increased interest in methods for leveraging crowds to scale data labeling efforts, and methods to learn from noisy crowd-sourced labels. The need to scale labeling is acute but particularly challenging in medical applications like pathology, due to the expertise required to generate quality labels and the limited availability of qualified experts. In this paper we investigate the application of Scalable Variational Gaussian Processes for Crowdsourcing (SVGPCR) in digital pathology. We compare SVGPCR with other crowdsourcing methods using a large multi-rater dataset where pathologists, pathology residents, and medical students annotated tissue regions breast cancer. Our study shows that SVGPCR is competitive with equivalent methods trained using gold-standard pathologist generated labels, and that SVGPCR meets or exceeds the performance of other crowdsourcing methods based on deep learning. We also show how SVGPCR can effectively learn the class-conditional reliabilities of individual annotators and demonstrate that gaussian-process classifiers have comparable performance to similar deep learning methods. These results suggest that SVGPCR can meaningfully engage non-experts in pathology labeling tasks, and that the class-conditional reliabilities estimated by SVGPCR may assist in matching annotators to tasks where they perform well.

[*] Corresponding author.

This work was supported by the Agencia Estatal de Investigación of the Spanish Ministerio de Ciencia e Innovación under contract PID2019-105142RB-C22/AEI/10.13039/501100011033, and the United States National Institutes of Health National Cancer Institute grants U01CA220401 and U24CA19436201. P.M. contribution was mostly before joining Microsoft Research, when he was supported by La Caixa Banking Foundation (ID 100010434, Barcelona, Spain) through La Caixa Fellowship for Doctoral Studies LCF/BQ/ES17/11600011.

Email address: `lee.cooper@northwestern.edu` (Lee A. D. Cooper)

1. Introduction

The amount of labeled data is one of the primary determinants of performance in machine learning applications, and the requirements of today's data-hungry algorithms have increased interest in scaling data labeling processes. A *crowdsourcing* approach that engages a broad set of individuals in labeling has been shown effective in tasks where expertise is not required such as labeling images in general categories [1, 2, 3]. In applications requiring expertise, sourcing labels from crowds is more challenging. Medical applications where labels are often assigned by expert diagnosticians with years of training are particularly difficult, but are arguably the applications where scaling is needed most due to the lack of availability of these experts and the clinical demands on their time [1, 4, 5]. Crowdsourcing in these scenarios can introduce significant tradeoffs between label volume and quality [4]. A more open process can generate more labels but may sacrifice quality. Engaging with more focused groups such as medical students that have some familiarity with the subject matter can improve quality and can enable some degree of vetting of participants.

Crowdsourced labeled data suffer from high label noise due to the different varying expertise degrees. One typical approach for obtaining reliable labeled data is the consensus, i.e., majority voting. However, in medical imaging, fixing/aggregating the noisy labels in a previous training step is not the best way. Instead, the best choice is to keep each annotation and model the expertise degree of each annotator. For example, weighting each annotation based on the annotator's reliability achieves this purpose [6]. Raykar et al. introduced a crowdsourcing model for classification with multiple annotators [7] based on logistic regression. This crowdsourcing framework jointly learns a latent classifier and annotators' reliability. This model was used for grading prostate cancer in tissue microarrays [8], where five different pathologists annotated each image. They estimated iteratively the classifier's coefficients and the annotators' reliability, following an Expectation-Maximization (EM) scheme. The logistic regression classifier overcame the inter-observer grading variability levels, and showed a good agreement with the participants. However, the flexibility of this model is limited, because it considers logistic regression as the latent classifier. An analogous crowdsourcing framework has been also used with more expressive classifiers such as deep neural networks [9, 10]. Gaussian processes were also introduced for crowdsourcing with sound results across different domains [11, 12, 13]. These models are Bayesian and non-parametric, making them suitable to learn good models without the need for very large labeled datasets. Also, they provide an accurate estimation of the uncertainty in the predictions [14].

In the dataset we will use in this paper, a group of medical students, pathology residents, and pathologists were organized to label tissue regions in digital pathology images of breast cancer specimens [15]. The average medical student may have some basic understanding of histology from their medical school coursework, but they will not have specific knowledge of histologic patterns in breast cancer [16]. The varied experience of these participants was leveraged to optimize effort while preserving quality. Medical students performed the majority of labeling tasks under the supervision of residents and attending pathologists, and feedback was provided openly via a Slack communication channel to avoid answering redundant questions. This significantly improved the quality of work that was given final review by pathologists, minimizing their work and interventions. While this process was effective, it worked because there was prior knowledge of participant experience, and it still required significant involvement of pathologists. This study set a high standard for quality for compatibility with learning algorithms that may not tolerate label noise well. A more tolerant algorithm would allow relaxation of these standards, enabling engagement of a broader audience without prior knowledge of their experience, and would require less oversight and review of their work. An ideal learning algorithm would be able to estimate the strengths and weaknesses of an individual participant during labeling, and to assign them examples accordingly to maximize efficiency [17].

In this paper we investigate how Scalable Gaussian Processes (SVGP) can learn from noisy crowdsourced labels in digital pathology applications (Figure 1). We explore a previously developed technique, SVGP for Crowdsourcing (SVGPCR), that learns how to infer accurate labels by estimating class-conditional reliabilities for individual annotators [18]. SVGPCR can learn these reliabilities from sparsely annotated datasets where each sample is labeled by only a subset of the annotators. The probabilistic modeling used by SVGPCR is described in detail in Methods.

We applied SVGPCR to a dataset where practicing pathologists, pathology residents, and medical students annotated breast cancer tissue regions. Our experiments found that SVGPCR trained on the noisy labels from non-experts is competitive with an equivalent SVGP trained using gold-standard expert labels. We also demonstrate how the learned annotator reliabilities accurately capture the class-conditional performance of individual annotators. We describe limitations of this approach and discuss how these approaches could be used to improve data labeling in digital pathology applications in the future. The code is publicly available at `https://github.com/wizmik12/crowdsourcing-digital-pathology-GPs`.

2. Methods

The data used in our experiments originate from an international study where pathology experts and non-experts annotated breast cancer tissue regions in a crowdsourcing process [15]. In this study a web-based platform was used to annotate breast cancer tissue regions by two senior/practicing pathologists (SP), and 20 non-pathologists (NP) consisting of medical students and fresh graduates. A study coordinator selected 161 rectangular regions of interest (ROIs) from 151 whole-slide images of formalin-fixed paraffin embedded sections from the TCGA Breast Cancer cohort. ROIs were selected to capture representative patterns of tumor, stroma, and immune infiltrates, as well as less common regions and structures including necrosis, blood vessels, and fat. Images and ROIs were hosted on a Digital Slide Archive server where participants could access them through a web-browser and use their mouse to annotate tissue regions in the ROIs using the polyline tool.

ROIs were assigned to two categories to provide both adequate breadth for training ML algorithms and to enable assessment of interobserver variability in annotation. Core ROIs provide breadth, being present in all 151 slides, and were divided among the users (approximately 6 per user) based on a difficulty score assigned by the study coordinator. Participants first annotated their core ROIs and then solicited feedback from an SP who applied corrections in multiple feedback cycles. This provided two versions of the core ROI: 1) Uncorrected core ROIs and 2) Corrected core ROIs. Ten additional Evaluation ROIs were created in the slide set and assigned to all NP participants to assess interobserver variability. Annotation of evaluation ROIs was performed following completion of core ROIs; evaluation ROI annotations were not corrected. The DICE coefficient for segmentation annotations made by SPs was as follows: 0.87 (tumor), 0.81 (stroma), and

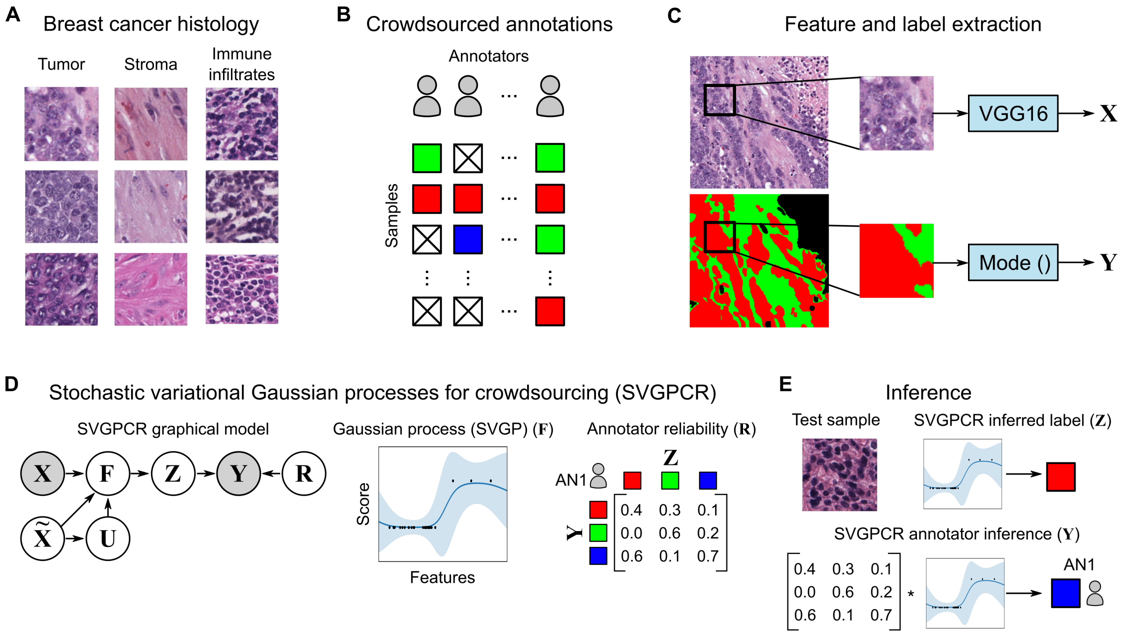


Figure 1: **Scalable variational Gaussian processes for crowdsourcing (SVGPCR) in digital pathology.** **(A)** This paper uses classification of predominant tissue patterns in breast cancer to investigate how SVGPCR can be used in crowdsourcing annotations for digital pathology. **(B)** The data used in this paper originates from a study where participants delineated tissue regions to produce semantic segmentation annotations in a set of curated Regions of Interest (ROI) (see Figure 2). SVGPCR enables a sparse study where most ROIs are not annotated by all participants. **(C)** To leverage SVGPCR in this application, we analyze patches from the annotated ROIs. Patches were selected where at least 50% of the pixels correspond to a single label. For each patch with a majority label $\mathbf{Y}$ we used VGG16 to extract a 512-dimensional feature vector $\mathbf{X}$ for SVGPCR training. **(D)** In SVGPCR, the observed annotation $\mathbf{Y}$ depends on the true label $\mathbf{Z}$ and annotator reliability $\mathbf{R}$. The scalable variational Gaussian process (SVGP) classifier $\mathbf{F}$ is trained to predict the true label from the features $\mathbf{X}$. $\tilde{\mathbf{X}}$ and $\mathbf{U} = \mathbf{F}(\tilde{\mathbf{X}})$ are used to improve the scalability of training (in GP terminology, they are called *inducing locations* and *inducing points* respectively, see Details on the machine learning algorithm). **(E)** Given a test patch, the SVGP classifier $\mathbf{F}$ can be used to infer the true label $\mathbf{Z}$, or combined with the reliability matrix of a specific annotator to infer how that annotator would label the patch.

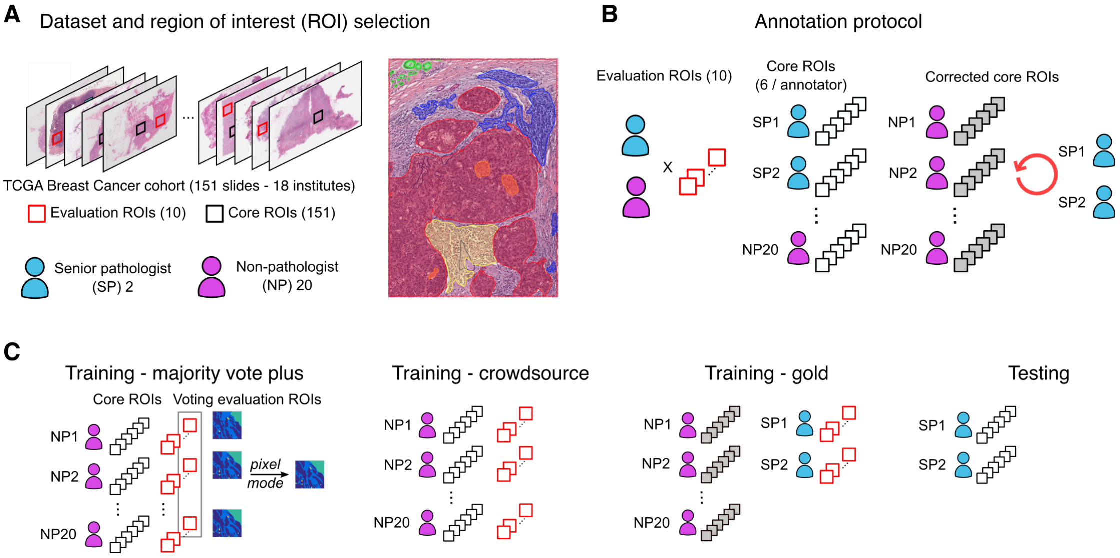


Figure 2: **Experimental design.** Our experiments combine annotations generated by experts (SP) and novice (NP) participants in a crowdsourcing study of breast cancer digital pathology images. **(A) 161 regions of interest in 151 slides were selected for inclusion in the annotation study [15]. 10 ROIs were selected as the Evaluation ROIs (red) and annotated by all participants. The remaining 151 ROIs were each assigned to individual annotators as Core ROIs (black). (B)** Participants used a web-based interface to annotate a number of tissue regions in each ROI including tumor, stroma, immune infiltration, and others. Core ROIs annotated by NPs were reviewed and corrected independently by either SPs, giving us paired uncorrected (black) and corrected gold standard (gray-filled) annotations. Annotations on Evaluation ROIs did not undergo correction. **(C)** We formed a number of training sets to assess various conditions. A "majority vote" (MV) training set smooths the labels over the evaluation set ROIs for assessing non-crowdsourcing methods. These are combined with the uncorrected core ROI annotations to increase data volume. A "crowdsource" (CR) dataset combines the uncorrected core and evaluation ROIs for NPs to form a training dataset with noisy labels for assessing crowdsourcing methods. A gold standard training dataset combines corrected ROIs from NPs with evaluation ROIs from the SPs. The testing set used to assess performance was composed of core ROIs from SPs and corrected core ROIs from NPs.

0.52 (lymphocytic infiltration). Further details on the interobserver variability for both SPs and NPs is discussed in detail in [15].

We performed a collection of experiments to assess the impact of training data quality and the effectiveness of crowdsourcing approaches. We considered a multiclass problem with three different classes: tumor, stroma, and immune infiltrates. We also compared Gaussian processes (with features from pre-trained convolutional networks) with state-of-the-art deep learning models like CrowdLayer [18, 10]. Data quality was examined by formulating three training sets with varying label quality (see Figure 2): 1. Gold standard training combines corrected core ROI annotations with SP annotations on evaluation ROIs; 2. Majority vote training (MV) combines uncorrected NP core ROI annotations with pixel-wise majority voting over NP evaluation ROI annotations; 3. Crowdsourcing training (CR) combines all uncorrected NP core ROI annotations and all NP evaluation ROI annotations. The gold standard training set represents a gold-standard where all annotations are generated, corrected, or approved by SPs. The MV training set represents a naive approach to improving data quality by averaging over noisy NP annotations. The CR training set represents a true crowdsourcing experiment where NP annotations are not corrected or revised by experts or smoothed through averaging.

First we measured the impact of training data quality on SVGP and VGG16 methods that weigh all labels and annotators equally, comparing their performance with smoothed label MV training and gold standard training. Next, we assessed the ability of crowdsourcing methods like AggNet [9], CrowdLayer (CL) [10], and SVGPCR [18], which learn annotator reliability using CR training generated through crowdsourcing with non-experts. The first two are recent methods based on deep learning. For Crowdlayer, depending on the annotator modeling, we can distinguish three different models: CL-MW, CL-VW, and CL-VWB. CL-VW incorporates a vector of per-class weights, an additional bias is considered for CL-VWB and, the most complex, CL-MW computes the whole confusion matrix of the annotators. SVGPCR is based on scalable Gaussian Processes.

Finally, we assessed the ability of SVGPCR to infer predictions from a specific annotator that reflect that annotator's class-conditional reliabilities. For these experiments we modified the CR training, reserving half of the evaluation ROIs for testing, and training the SVGPCR on the uncorrected NP core ROIs and the remaining evaluation ROIs. SVGPCR inference was performed for each annotator and evaluation ROI in the testing set and compared to the annotations of that annotator using the DICE coefficient. Dense predictions were generated in these experiments using sliding windows with 95% overlap to enable visual comparison.

Here we describe the formulation of a scalable SVGPCR algorithm that can learn from sparsely annotated datasets. Additional details are presented in the Supplementary Information and in the SVGPCR paper [18]. The inputs for training an SVGPCR model are the features $\mathbf{X}$, that are derived from the images, and the crowdsourced labels $\mathbf{Y}$. SVGPCR simultaneously learns both a classification model and the class-conditional reliabilities for each annotator. First, an underlying Gaussian Process (GP) model is learned to classify previously unobserved samples. The GP is denoted by $\mathbf{F}$ in Figure 3 ($\mathbf{U}$ and $\tilde{\mathbf{X}}$ are the inducing points and the inducing point locations respectively, and they are introduced for scalability). Second, the reliabilities of each annotator are modeled using per-annotator confusion matrices that describes the reliabilities of each annotator in labeling each class ($\mathbf{R}$ in Figure 3). Both $\mathbf{F}$ and $\mathbf{R}$ are connected by the variable $\mathbf{Z}$, which represents the unknown true labels of the training samples. This unknown

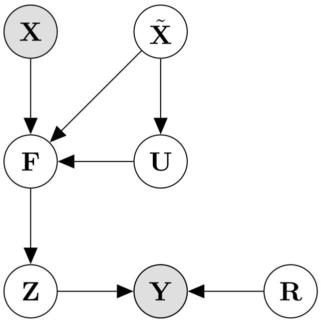


Figure 3: Probabilistic graphical model for SVGPCR. Dark variables refer to observed variables while light variables refer to latent variables (to be estimated). The observed variables are the features $\mathbf{X}$ and the annotations $\mathbf{Y}$ made by several annotators. The annotations depend on the true labels $\mathbf{Z}$ and the reliability of the annotators, $\mathbf{R}$. The true labels are modeled by latent variables $\mathbf{F}$ with a GP prior. Once the training is finished, the latent classifier can predict the true label on unseen samples. For scalability, $\tilde{\mathbf{X}}$ and $\mathbf{U}$ summarize data information lightening the computational cost ($\tilde{\mathbf{X}}$ is much smaller than $\mathbf{X}$).

variable is integrated out and estimated during training jointly with the classifier $\mathbf{F}$ and reliabilities $\mathbf{R}$.

This work addresses a K-class classification problem with crowdsourced labels. The training set consists of N instances $\{(\mathbf{x}_n, \mathbf{y}_n^a): \ n = 1, \ldots, N; \ a \in A_n\}$, where $\mathbf{x}_n \in \mathbb{R}^D$ is the feature vector of the n-th instance, and $\mathbf{y}_n^a$ is the label provided by the a-th annotator for the n-th instance. We represent labels as one-hot encoded vectors, i.e., the k-th class is specified by a vector in which all elements are zeros except for a single one in the k-th position. The matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\intercal \in \mathbb{R}^{N \times D}$ contains the features of all the training instances and the set of all the annotations is defined as $\mathbf{Y} = \{\mathbf{y}_n^a : n = 1, \ldots, N, a \in A_n\}$ where A_n is the subset of annotators that labeled the n-th instance. Note that each sample can be annotated by a different subset of annotators.

In this approach, each instance is assumed to have an (unknown) true label, $\mathbf{z}_n \in \{\mathbf{e}_1, \ldots, \mathbf{e}_K\}$. The reliability of each annotator is modeled by a confusion matrix $\mathbf{R}^a = (r_{ij}^a)_{1 \le i,j \le K}$. Each row of this matrix represents the label provided by the a-th annotator, and each column the true class. Notice that it is normalized, so each column adds up to 1, and the elements represent conditional probabilities. In other words, $\mathrm{p}(\mathbf{y}^a = \mathbf{e}_i | \mathbf{z} = \mathbf{e}_j) = r_{ij}^a$. Notice that the reliability matrix of a perfect annotator will be the identity. Mathematically, this is given by

$$\mathrm{p}(\mathbf{y}_n^a | \mathbf{z}_n, \mathbf{R}^a) = [\mathbf{y}_n^a]^\intercal \mathbf{R}^a \mathbf{z}_n. \tag{1}$$

Assuming independence among annotators, we have

$$\mathrm{p}(\mathbf{Y}|\mathbf{Z}, \mathbf{R}) = \prod_{n=1}^{N} \prod_{a \in A_n} \mathrm{p}(\mathbf{y}_n^a | \mathbf{z}_n, \mathbf{R}^a), \tag{2}$$

where $\mathbf{Z} = \{\mathbf{z}_n : n = 1, \ldots, N\}$ and $\mathbf{R} = \{\mathbf{R}^a : a = 1, \ldots, A\}$ contain the true labels of all instances and the reliability matrices of all annotators, respectively.

SVGPCR defines a prior (independent) Dirichlet distribution over $\mathbf{R}$,

$$\mathrm{p}(\mathbf{R}) = \prod_{a=1}^{A} \prod_{j=1}^{K} \mathrm{p}(\mathbf{r}_j^a) = \prod_{a=1}^{A} \prod_{j=1}^{K} \mathrm{Dir}(\mathbf{r}_j^a | \alpha_{1j}^a, \ldots, \alpha_{Kj}^a), \tag{3}$$

where $\mathbf{r}_j^a = (r_{1j}^a, \ldots, r_{Kj}^a)^\intercal$ is the j-th column of $\mathbf{R}^a$. The hyperparameters $\boldsymbol{\alpha} = \{\alpha_{ij}^a : i, j = 1, \ldots, K,\ a = 1, \ldots, A\}$ of the prior distribution allow for including assumptions on the reliability of the annotator. When there is no prior knowledge about the annotators' behavior, the most common choice is to use a non-informative uniform distribution, i.e., $\alpha_{ij}^a = 1$.

So far, we have seen how SVGPCR models the crowdsourced annotations given the true labels. Now, we model the relationship between the true labels $\mathbf{Z}$ and the features $\mathbf{X}$ by introducing a latent classifier based on stochastic variational Gaussian procesess [19]. That is, K latent variables $\mathbf{f}_{n,:} = \{f_k(\mathbf{x}_n)\}_{k=1}^K$ model the (unknown) true label $\mathbf{z}_n$ through a specific likelihood $\mathrm{p}(\mathbf{z}_n|\mathbf{f}_{n,:})$. The latent variables provide scores in $\mathbb{R}$ to each sample and the likelihood maps them to the $[0, 1]$ interval. We use the soft-max likelihood which is defined by

$$\mathrm{p}(\mathbf{z}_n = \mathbf{e}_k|\mathbf{f}_{n,:}) = \frac{e^{f_{n,k}}}{\sum_{c=1}^K e^{f_{n,c}}}. \tag{4}$$

To lighten the notation, we denoted the latent variables by $f_k(\mathbf{x}_n) = f_{n,k}$. Assuming that the class labels are independent given the latent variables, we factorize the likelihood across the different samples:

$$\mathrm{p}(\mathbf{Z}|\mathbf{F}) = \prod_{n=1}^N \mathrm{p}(\mathbf{z}_n|\mathbf{f}_{n,:}), \tag{5}$$

where $\mathrm{p}(\mathbf{z}_n|\mathbf{f}_{n,:})$ is given by eq. (4). $\mathbf{F}$ gathers the latent variables in a $N \times K$ matrix where $f_{n,k}$ is placed in the n-th row and k-th column. Notice that the K latent variables are in the columns, $\mathbf{f}_k$, and the rows gather the value of each variable for the N instances $\mathbf{f}_{n,:}$.

The latent variables $\{\mathbf{f}_k\}_{k=1}^K$ are modeled by independent GP priors. This imposes that $\{f_{n,k}\}_{n=1}^N$ follow a multivariate Gaussian distribution (for a fixed k). We also assume that this Gaussian distribution has $\mathbf{0}$ mean and the covariance matrix is given by a kernel function. In this work, we use the Squared Exponential (SE) kernel, which is defined by $k(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/(2l^2))$ [20]. Therefore, the prior over the latent variables $\mathbf{F}$ is given by

$$\mathrm{p}(\mathbf{F}|\boldsymbol{\Theta}, \mathbf{X}) = \prod_{k=1}^K \mathrm{p}(\mathbf{f}_k|\boldsymbol{\Theta}, \mathbf{X}) = \prod_{k=1}^K \mathcal{N}(\mathbf{f}_k|\mathbf{0}, \mathbf{K}_{\mathbf{XX}}), \tag{6}$$

where $\boldsymbol{\Theta}$ includes σ and l (i.e., the kernel hyperparameters), and the covariance matrix is $\mathbf{K}_{\mathbf{XX}} = \mathbf{K}(\mathbf{X}, \mathbf{X}) = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j}$. Notice that the SE kernel is very expressive and performs remarkably well in different scenarios [20]. In particular, it encodes desirable properties in the covariance matrix, such as smoothness.

In summary, we have defined the following probabilistic model:

$$\mathrm{p}(\mathbf{Y}, \mathbf{Z}, \mathbf{F}, \mathbf{R}|\boldsymbol{\Theta}) = \underbrace{\mathrm{p}(\mathbf{Y}|\mathbf{Z}, \mathbf{R})\mathrm{p}(\mathbf{R})}_{\text{CR modelling}} \underbrace{\mathrm{p}(\mathbf{Z}|\mathbf{F})}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{F}|\mathbf{X}, \boldsymbol{\Theta})}_{\text{GP prior}}. \tag{7}$$

This model is not scalable because standard GPs involve the inversion of an $N \times N$ dimensional matrix. To overcome this limitation and deal with large datasets the sparse approximation is used [19]. This approximation introduces $M \ll N$ inducing points. These inducing points summarize the information of the observations and will lighten

	F1 score	Accuracy	Log loss	AUC
VGG-gold	0.8088	0.8440	0.7073	0.9271
VGG-MV	0.7975	0.8325	0.6635	0.9201
SVGP-gold	**0.8157**	**0.8582**	**0.3938**	**0.9373**
SVGP-MV	0.7919	0.8458	0.4261	0.9289
SVGPCR	0.8147	0.8579	0.3983	0.9360

Table 1: Performance on the test set: F1 score, accuracy, log loss, and AUC values. Gold refers to expert labels, MV to majority vote labels, SVGPCR to crowdsource labels.

	F1 score	Accuracy	Log loss	AUC
AggNet [9]	0.7998	0.8433	0.6814	0.9287
CL-MW [10]	0.8158	0.8570	0.4963	0.9317
CL-VW [10]	0.8072	0.8421	0.4911	0.9264
CL-VWB [10]	**0.8179**	0.8554	0.5536	0.9301
SVGPCR [18]	0.8147	**0.8579**	**0.3983**	**0.9360**

Table 2: Performance of crowdsourcing methods on the test set: F1 score, accuracy, log loss, and AUC values. These methods use non-expert labels.

the computational cost. They are values of the GP function. Notice that the inducing locations, where the GP is valued to compute the inducing points, may not be instances of the training set. We denote by $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_M]^\intercal \in \mathbb{R}^{M\times D}$ the inducing locations while $\mathbf{U}$ corresponds to their value after the GP is applied. In other words, $\mathbf{U}$ is the evaluation of the GP on $\tilde{\mathbf{X}}$, just like $\mathbf{F}$ is on $\mathbf{X}$. Importantly, the locations $\tilde{\mathbf{X}}$ are optimized during training. Finally, the sparse probabilistic model is given by

$$\mathrm{p}(\mathbf{Y}, \mathbf{Z}, \mathbf{F}, \mathbf{U}, \mathbf{R}|\boldsymbol{\Theta}) = \underbrace{\mathrm{p}(\mathbf{Y}|\mathbf{Z}, \mathbf{R})\mathrm{p}(\mathbf{R})}_{\text{CR modelling}}\ \underbrace{\mathrm{p}(\mathbf{Z}|\mathbf{F})}_{\text{likelihood}}\ \underbrace{\mathrm{p}(\mathbf{F}|\mathbf{U}, \boldsymbol{\Theta})\mathrm{p}(\mathbf{U}|\boldsymbol{\Theta})}_{\text{GP prior}}. \tag{8}$$

Once the probabilistic model is defined, the posterior distribution $\mathrm{p}(\mathbf{Z}, \mathbf{F}, \mathbf{U}, \mathbf{R}|\mathbf{Y}, \boldsymbol{\Theta})$ must be computed. Since this cannot be achieved in closed-form (integrating out $\mathbf{Z}$ in (8) is intractable), SVGPR resorts to variational inference. The mathematical details for the variational inference step and for the predictive distribution are provided in the supplementary material.

3. Results

Table 1 depicts the performance of the SVGP and VGG methods with the different training sets. We found that training data quality impacts the performance of the SVGP and VGG methods. Training on the gold standard data resulted in improvements in F1 score, AUC, and accuracy for both SVGP and VGG when compared with MV training. For SVGP the gold standard training data improved the F1 score by 3.0% to 0.816. Similar improvements were observed for AUC (0.9% increase to 0.973), and accuracy (1.5% increase to 0.858). For VGG the gold standard training data improved the F1 score by 1.4% to 0.809. Similar improvements were observed for AUC (0.7% increase to 0.927), and accuracy (1.3% increase to 0.844). For log loss we observed an improvement for SVGP (7.6% reduction to 0.3938) but for VGG the loss increased (6.5% increase to 0.7073). Comparing SVGP and VGG with gold standard training we observed a small

DICE	Tumor	Stroma	Immune Infiltrates	Overall
Ground truth	0.8529	0.7979	0.6905	0.8072
Participant's behavior	0.8132 ± 0.0342	0.7286 ± 0.0392	0.4841 ± 0.1310	0.7789 ± 0.0237

Table 3: DICE values for participant's behavior and ground-truth (i.e., expert annotation) predictions. The results are computed per-class and globally. Furthermore, confidence intervals of 95% are computed for the 20 participants.

performance benefit for SVGP with a slightly higher F1 score (0.8% increase), AUC (1.0% increase), accuracy (1.7% increase), and lower loss (44% reduction) than VGG.

Table 2 depicts the performance of different crowdsourcing methods trained with the CR training set. CrowdLayer and SVGPCR have similar performance, with SVPGCR having a slight advantage in AUC, accuracy, and loss. CrowdLayer-VWB had a small advantage in F1 score (0.4% increase to 0.818), where SVGPCR had an advantage over the next best CrowdLayer method in AUC (0.4% higher than CL-MW), accuracy (0.1% higher than CL-MW), and loss (18.9% lower than CL-MW). AggNet has the lowest performance of crowdsourcing methods in all metrics except for accuracy. The best performing crowdsourcing methods were competitive with SVGP and VGG with gold standard training. SVGPCR trained on noisy CR labels is very similar to SVGP trained with gold standard labels with both methods having similar F1 scores (0.815 versus 0.816), AUCs (0.936 versus 0.937), accuracies (0.858 for both), and losses (0.398 versus 0.393). These differences are small when compared to differences between SVGP with MV training and SVGP with gold standard training.

Figure 4 shows examples of inferred predictions for individual annotators. Visual inspection of these predictions shows that SVGPCR can learn and reproduce the biases of individual annotators. NP17 tends to call some stromal regions as tumor, and the SVGPCR inferred predictions for NP17 also exhibit this tendency. NP19 is less sensitive in annotating tumor, missing a large region that was annotated by the SP, and we see this same lack of sensitivity in SVGPCR inference for NP19. NP21 is not sensitive in detecting a group of inflammatory cells, and we also see that their SVGPCR inference lacks sensitivity in detecting these cells as well. Quantitative analysis of agreement between SVGPCR inferences for specific annotators and their uncorrected annotations is presented in Table 3. The quantization is made by reconstructing the pixel-level of annotators using the patches annotations. The similarity of the annotations and the predictions is performed using the DICE coefficient. This coefficient measures the similarity between them. The 95% confidence interval of the DICE scores averaged over the 20 NPs is 0.7789 ± 0.0237. The average DICE score when comparing SVGPCR inferred gold standard with the expert SP annotations lies outside this interval at 0.8072.

4. Discussion

Data is often the limiting factor in training and validating machine learning algorithms for biomedical applications. When domain experts like pathologists are needed to produce ground-truth labels, generating data at the scale required by algorithms like convolutional networks is often difficult. This study seeks to address this problem by examining how a probabilistic approach to integrating annotations from novices can compete with algorithms trained using gold-standard data generated by experts. As a statistical machine learning method, Gaussian processes provide a framework for estimating the accuracy of annotators, including class-conditional accuracies, and to use this

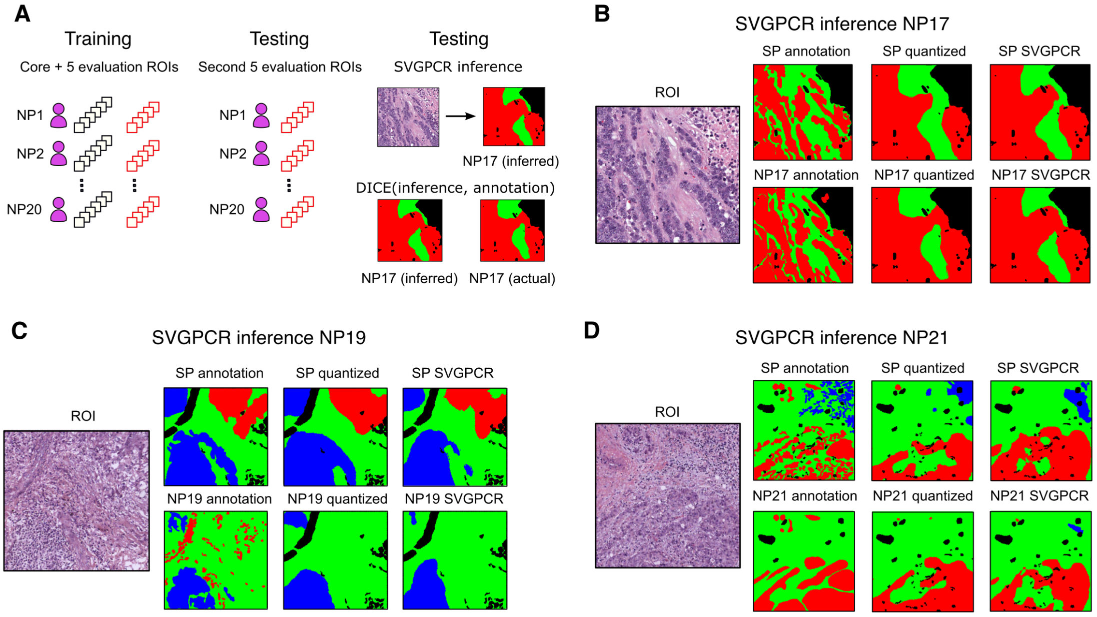


Figure 4: **Visualizing annotator-specific inferences.** We performed additional experiments to assess the ability of SVGPCR to learn the biases of individuals. The color in the masks encode tumor (red), stroma (green), lymphocytic infiltrates (blue) and other classes (black). **(A)** Two SVGPCR classifiers were trained. The first training set combined the core ROIs and first 5 evaluation ROIs, and performed inference on the second 5 evaluation ROIs. The evaluation ROIs were then swapped, and the training and inference were repeated. For each ROI, the trained SVGP and reliability annotation matrices were used to generate an annotator-specific inference. This inference was compared with the actual annotation and the annotation from the SP to observe differences. The patch-based analysis resulted in some quantization, so the quantized and original annotations are both presented. **(B)** This ROI contains a band of stroma from the upper center to the lower right that separates two regions of tumor, and a region of necrosis on the right. The inferred true labels correspond closely to the SP annotation. Participant NP17 is more sensitive in annotating tumor, and their inferred annotation exhibits the same pattern. **(C)** This ROI contains an island of tumor separated from regions of dense immune infiltrates by a wide area of stroma. The inferred true labels correspond closely with the SP annotation. Participant NP19 is not very sensitive in labeling tumor by comparison, and the tumor in the annotator inference is also absent. **(D)** This ROI contains tumor in the lower left and a small pocket of immune infiltrates in the upper right. The immune infiltrates are present in both the SP annotation and the inferred true labels. The immune infiltrates are absent from the annotation of participant NP21, and are mostly absent from the inferred annotation.

information in making inferences of ground truth. Our experiments show that SVGPCR trained on noisy labels obtained from novices in digital pathology crowdsourcing studies can compete with state of the art algorithms trained on gold standard labels.

We used a unique data resource to compare Gaussian processes based methods with other crowdsourcing approaches. The BRCA tissue region dataset contains over 20,000 tissue regions, including both novice and expert-corrected annotations, enabling comparison of crowdsourcing methods trained on novice annotations to methods trained on gold-standard annotations. Our experiments demonstrated that data quality impacts the performance of methods that are not based on crowdsourcing. SVGP and VGG models trained using a "majority vote" training dataset that averaged novice annotations had inferior performance compared to the same models trained using gold standard annotations. Under the optimistic conditions of training with gold standard annotations, SVGP and VGG had similar performance, with SVGP having a slight advantage in F1, AUC, accuracy and a large improvement in loss on the testing data, showing that Gaussian process models can compete with convolutional networks in this example.

The best crowdsourcing methods including SVGPCR and CrowdLayer variants trained using novice annotations have performance comparable to methods trained using gold standard annotations. This result suggests that in some circumstances, expert correction of novice annotations may not be necessary for annotations used in training. Performance differences for SVGPCR and CrowdLayer were small compared to differences between methods trained with majority vote and gold standard data, suggesting that the annotator and class conditional weighting applied by crowdsourcing methods is superior to basic smoothing of novice data labels. SVGPCR performance in classifying tumor and stroma was significantly higher than for immune infiltrates. This parallels the patterns of interobserver variability observed during the crowdsourcing study. Tumor and stroma are defined by sharp boundaries and in our annotation data we see significantly better concordance among annotators for these tissue types. Immune infiltration is diffuse and regions infiltrated by immune cells lack a sharp boundary, requiring annotators to judge their density which is much more subjective. This translates to higher interobserver variability among annotators for immune infiltrates, and likely presents a greater challenge for SVGPCR. Regions of immune infiltration are also less prevalent in our dataset than regions of tumor and stroma.

We also showed how SVGPCR can reproduce the biases of specific annotators through inference. This result suggests that SVGPCR could help assigning work to annotators on the basis of their relative strengths and weaknesses as observed in their class-conditional accuracies. By modeling class-conditional annotator accuracy, SVGPCR learns how to weight the labels of each annotator during training to improve inference of gold standard labels. We provide visual and quantitative evidence that show how annotator-specific inferences produced by SVGPCR agrees with the withheld annotations on these test images, and reflects the sensitivities of annotators to various classes.

While these results suggest that SVGPCR may help reduce the annotation burden in digital pathology tasks, there are some important limitations in our study. Quantizing segmentation annotations to the patch level was necessary to provide a neighborhood of pixels for SVGPCR to learn from, however, this results in a loss of detail. While this quantization was necessary to conduct our studies, SVGPCR may be more appropriate for patch level problems like cell classification than for segmentation problems where fine details need to be represented. While SVGPCR likely benefits from the presence of a variety of annotators, some being more specific or more sensitive for different classes, it is not well understood when variability in annotations may pose a problem for learning. Furthermore, while some common evaluations regions among annotators are likely necessary for SVGPCR to learn the strengths and weaknesses of each annotator, it is not well understood how the balance of evaluation and core ROIs impacts SVGPCR performance. The core regions increase the breadth of the training set, and the annotation of evaluation regions reduces this breadth given a fixed budget of annotator time. We also plan to explore how the class-conditional accuracies learned by SVGPCR can improve assignment data to participants in crowdsourcing experiments and can help participants to understand their weaknesses and to improve them. This could be accomplished by iterative training of an SVGPCR model during crowdsourcing studies. We are also interested in exploring how the number of evaluation and core regions impacts SVGPCR performance.

References

[1] A. Kovashka, O. Russakovsky, L. Fei-Fei, Crowdsourcing in Computer Vision, Now Publishers Inc., Hanover, MA, USA, 2016.

[2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 740–755.

[3] E. Saralioglu, O. Gungor, Crowdsourcing in remote sensing: A review of applications and future directions, IEEE Geoscience and Remote Sensing Magazine 8 (4) (2020) 89–110. doi:10.1109/MGRS.2020.2975132.

[4] H. Irshad, L. Montaser-Kouhsari, G. Waltz, O. Bucur, J. A. Nowak, F. Dong, N. W. Knoblauch, A. Beck, Crowdsourcing image annotation for nucleus detection and segmentation in computational pathology: evaluating experts, automated methods, and the crowd., Pacific Symposium on Biocomputing. (2015) 294–305.

[5] S. Ørting, A. Doyle, A. van Hilten, M. Hirth, O. Inel, C. R. Madan, P. Mavridis, H. Spiers, V. Cheplygina, A survey of crowdsourcing in medical image analysis, arXiv preprint arXiv:1902.09159 (2019).

[6] D. Karimi, H. Dou, S. K. Warfield, A. Gholipour, Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis, Medical Image Analysis 65 (2020) 101759. doi:https://doi.org/10.1016/j.media.2020.101759.
URL http://www.sciencedirect.com/science/article/pii/S1361841520301237

[7] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, L. Moy, Learning from crowds, Journal of Machine Learning Research 11 (43) (2010) 1297–1322.
URL http://jmlr.org/papers/v11/raykar10a.html

[8] G. Nir, S. Hor, D. Karimi, L. Fazli, B. F. Skinnider, P. Tavassoli, D. Turbin, C. F. Villamil, G. Wang, R. S. Wilson, K. A. Iczkowski, M. S. Lucia, P. C. Black, P. Abolmaesumi, S. L. Goldenberg, S. E. Salcudean, Automatic grading of prostate cancer in digitized histopathology images: Learning from multiple experts, Medical Image

Analysis 50 (2018) 167 – 180. doi:https://doi.org/10.1016/j.media.2018.09.005.
URL `http://www.sciencedirect.com/science/article/pii/S1361841518307497`

[9] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, N. Navab, Aggnet: Deep learning from crowds for mitosis detection in breast cancer histology images., IEEE Trans. Med. Imaging 35 (5) (2016) 1313–1321.
URL `http://dblp.uni-trier.de/db/journals/tmi/tmi35.html#AlbarqouniBABDN16`

[10] F. Rodrigues, F. Pereira, Deep learning from crowds, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), AAAI Press, 2018, pp. 1611–1618.

[11] F. Rodrigues, F. Pereira, B. Ribeiro, Gaussian process classification and active learning with multiple annotators, in: E. P. Xing, T. Jebara (Eds.), Proceedings of the 31st International Conference on Machine Learning, Vol. 32 of Proceedings of Machine Learning Research, PMLR, Bejing, China, 2014, pp. 433–441.
URL `http://proceedings.mlr.press/v32/rodrigues14.html`

[12] P. Ruiz, P. Morales-Álvarez, R. Molina, A. K. Katsaggelos, Learning from crowds with variational Gaussian processes, Pattern Recognition 88 (2019) 298 – 311. doi:https://doi.org/10.1016/j.patcog.2018.11.021.
URL `http://www.sciencedirect.com/science/article/pii/S0031320318304060`

[13] P. Morales-Álvarez, P. Ruiz, R. Santos-Rodríguez, R. Molina, A. K. Katsaggelos, Scalable and efficient learning from crowds with Gaussian processes, Information Fusion 52 (2019) 110 – 127. doi:https://doi.org/10.1016/j.inffus.2018.12.008.
URL `http://www.sciencedirect.com/science/article/pii/S1566253518304664`

[14] C. Rasmussen, C. Williams, Gaussian Processes for Machine Learning, Adaptive Computation and Machine Learning, MIT Press, Cambridge, MA, USA, 2006.

[15] M. Amgad, H. Elfandy, H. Hussein, L. A. Atteya, M. A. T. Elsebaie, L. S. Abo Elnasr, R. A. Sakr, H. S. E. Salem, A. F. Ismail, A. M. Saad, J. Ahmed, M. A. T. Elsebaie, M. Rahman, I. A. Ruhban, N. M. Elgazar, Y. Alagha, M. H. Osman, A. M. Alhusseiny, M. M. Khalaf, A.-A. F. Younes, A. Abdulkarim, D. M. Younes, A. M. Gadallah, A. M. Elkashash, S. Y. Fala, B. M. Zaki, J. Beezley, D. R. Chittajallu, D. Manthey, D. A. Gutman, L. A. D. Cooper, Structured crowdsourcing enables convolutional segmentation of histology images, Bioinformatics 35 (18) (2019) 3461–3467. arXiv:https://academic.oup.com/bioinformatics/article-pdf/35/18/3461/30024472/btz083.pdf, doi:10.1093/bioinformatics/btz083.
URL `https://doi.org/10.1093/bioinformatics/btz083`

[16] M. Sadofsky, B. Knollmann-Ritschel, R. M. Conran, M. B. Prystowsky, National standards in pathology education: developing competencies for integrated medical school curricula, Arch. Pathol. Lab. Med. 138 (3) (2014) 328–332.

[17] Y. Zheng, G. Li, Y. Li, C. Shan, R. Cheng, Truth inference in crowdsourcing: Is the problem solved?, Proc. VLDB Endow. 10 (5) (2017) 541–552.

doi:10.14778/3055540.3055547.
URL `https://doi.org/10.14778/3055540.3055547`

[18] P. Morales-Álvarez, P. Ruiz, S. Coughlin, R. Molina, A. K. Katsaggelos, Scalable variational Gaussian processes for crowdsourcing: Glitch detection in LIGO, IEEE Transactions on Pattern Analysis and Machine Intelligence (2020).

[19] J. Hensman, A. G. de G. Matthews, Z. Ghahramani, Scalable variational Gaussian process classification, in: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015, 2015.
URL `http://jmlr.org/proceedings/papers/v38/hensman15.html`

[20] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2006.

Chapter 5

Multiple Instance Learning in CT scans using Deep Gaussian Processes

5.1 Publication details

Authors: Miguel López-Pérez, Arne Schmidt, Yunan Wu, Rafael Molina, Aggelos K. Katsaggelos.
Title: Deep Gaussian Processes for Multiple Instance Learning: Application to CT Intracranial Hemorrhage Detection.
Publication: Computer Methods and Programs in Biomedicine, vol. 219, 106783-106783, June 2022.
Status: Published.
Quality indices:

- Impact Factor (JCR 2020): 5.428.
- Rank: 13/110 (Q1) in Computer Science, Theory & Methods.

5.2 Main contributions

- We propose a new probabilistic Multiple Instance Learning model based on Deep Gaussian Processes which we name DGPMIL. We introduce for the first time DGPs to the MIL problem and study their application to this problem.
- We study the behavior of the DGPMIL on a controlled experiment on MNIST.
- We apply DGPMIL combined with an attention-based CNN to Intracranial Hemorrhage Detection. This application studies the suitability of DGPs to ICH detection with labels only at scan level. We compare our approach to other state-of-the-art methods for this problem based on CNNs and shallow GPs.

Deep Gaussian Processes for Multiple Instance Learning: Application to CT Intracranial Hemorrhage Detection

Miguel López-Pérez[a,*], Arne Schmidt[a], Yunan Wu[b], Rafael Molina[a], Aggelos K. Katsaggelos[b]

[a]*Department of Computer Science and Artificial Intelligence, University of Granada, 18010 Granada, Spain.*
[b]*Department of Electrical Computer Engineering, Northwestern University, Evanston, IL, 60208 USA.*

Abstract

Background and objective:
Intracranial hemorrhage (ICH) is a life-threatening emergency that can lead to brain damage or death, with high rates of mortality and morbidity. The fast and accurate detection of ICH is important for the patient to get an early and efficient treatment. To improve this diagnostic process, the application of Deep Learning (DL) models on head CT scans is an active area of research. Although promising results have been obtained, many of the proposed models require slice-level annotations by radiologists, which are costly and time-consuming.
Methods:
We formulate the ICH detection as a problem of Multiple Instance Learning (MIL) that allows training with only scan-level annotations. We develop a new probabilistic method based on Deep Gaussian Processes (DGP) that is able to train with this MIL setting and accurately predict ICH at both slice- and scan-level. The proposed DGPMIL model is able to capture complex feature relations by using multiple Gaussian Process (GP) layers, as we show experimentally.
Results:
To highlight the advantages of DGPMIL in a general MIL setting, we first conduct several controlled experiments on the MNIST dataset. We show that multiple GP layers outperform one-layer GP models, especially for complex feature distributions. For ICH detection experiments, we use two public brain CT datasets (RSNA and CQ500). We first train a Convolutional Neural Network (CNN) with an attention mechanism to extract the image features, which are fed into our DGPMIL model to perform the final predictions. The results show that DGPMIL model outperforms VGPMIL as well as the attention-based CNN for MIL and other state-of-the-art methods for this problem. The best performing DGPMIL model reaches an AUC-ROC of 0.957 (resp. 0.909) and an AUC-PR of 0.961 (resp. 0.889) on the RSNA (resp. CQ500) dataset.

[*] Corresponding author.

This work was supported by project P20_00286 funded by FEDER/Junta de Andalucía-Consejería de Transformación Económica, Industria, Conocimiento y Universidades and the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska Curie grant agreement No 860627 (CLARIFY Project).

Email addresses: mlopez@decsai.ugr.es (Miguel López-Pérez), arne@decsai.ugr.es (Arne Schmidt), yunanwu2020@u.northwestern.edu (Yunan Wu), rms@decsai.ugr.es (Rafael Molina), a-katsaggelos@northwestern.edu (Aggelos K. Katsaggelos)

Preprint submitted to Computer Methods and Programs in Biomedicine

Conclusion:
The competitive performance at slice- and scan-level shows that DGPMIL model provides an accurate diagnosis on slices without the need for slice-level annotations by radiologists during training. As MIL is a common problem setting, our model can be applied to a broader range of other tasks, especially in medical image classification, where it can help the diagnostic process.

Keywords: Multiple Instance Learning, Deep Gaussian Processes, Intracranial Hemorrhage Detection, Weakly Supervised Learning

1. Introduction

Intracranial hemorrhage is a severe life-threatening emergency with high rates of mortality and permanent disability. It is initially caused by blood leaking inside the cranium, where the rapidly increasing blood pressure of the brain leads to severe brain damage or death [1]. It is reported that around 40000 to 67000 subjects suffer from ICH per year in the United States [2] and 30% of them eventually die [3]. To avoid death or remaining damages, early treatment is crucial. The study shows that, without timely brain surgery, nearly half of the deaths occur in the first 24 hours and only 20% of the surviving patients have the chance to completely recover at the end [2], indicating the important role of a fast and accurate ICH diagnosis in improving the survival rates and chances of recovery. Computed Tomography (CT) is a widely used non-invasive imaging technique for the ICH diagnosis, that is accessible and cheap for patients and at the same time, convenient and fast for radiologists. However, studies show that radiologists may misdiagnose after long hours of CT scans readings [4, 5]. As Computer-aided diagnosis (CAD) methods can help to reduce the workload of radiologists and provide an accurate diagnosis, they are of high clinical importance.

With the rapid development of DL, several models have been proposed to detect ICH. CNNs foster self-learning filters to focus on regions of interest without the need for manual feature extractions. The simplest way is to apply DL models on a single slice directly and predict the ICH at slice-level. For instance, Phong et al. [6] compared three types of traditional CNN models and found that models with pre-trained weights on non-medical images improved the ICH diagnosis. Cho et al. [7] developed a cascade DL model based on CNNs and dual fully convolutional networks to improve the sensitivity in identifying ICH. Although these models achieved good classification performances, it is challenging to collect a large number of slice annotations because manual labeling is time-consuming and requires expert knowledge. The ground truth at scan-level is, however, relatively easy to obtain, as it can be generated directly from the clinical radiologists' report. Therefore, an emerging approach using only scan-level labels consists of predicting ICH on full 3D scans. For instance, Titano et al. [8] utilized a 3D Resnet-50 CNN to predict ICH on brain scans and Jnawali et al. [9] ensembled three different 3D CNNs to improve the detection rate of ICH. However, one major problem of 3D CNNs lies in their highly expensive computation, leading to out-of-memory errors during the training processes. In addition, 3D models are not able to indicate the specific slice that contains the possible ICH inside a scan. This is however crucial to facilitate the ICH localization.

Another approach that uses only scan-level labels is the MIL paradigm. MIL is a weakly-supervised learning method that has been proposed to solve the problems when only bag labels are available [10]. It has been applied in many medical domains. Campanella et al. [11] trained a MIL model to diagnose cancer in histopathological images with slide labels by finding the highest probability per bag and then applying a recurrent neural network on the extracted features of each instance to predict the whole slide. Recently, attention-based methods are gaining more and more popularity in the field of medical images for the MIL setting. Similar to channel attention mechanisms that are weighting each channel of a CNN layer with attention weights [12], the attention weights in the case of MIL are assigned to the instances [13]. These instance attention weights provide insight into the contribution of each instance to the bag predictions. Several approaches have extended this attention mechanism to different medical applications: Han et al. [14] proposed an attention-based deep 3D MIL to diagnose COVID-19 from chest CT, where the attention mechanism is able to find key instances to interpret the specific infection areas of COVID-19. Qi et al. [15] developed another deep represented MIL to classify COVID-19 from normal pneumonia, which was first pre-trained to generate each instance feature and then generated predictions using the k-nearest neighbor. Similarly, they found that the attention weights highlight infected lesions, providing strong evidence for the diagnosis. Other approaches for the MIL problem are based on Gaussian Processes (GPs). Gaussian Processes were first proposed as Variational Gaussian Processes for MIL (VGPMIL) obtaining promising results in many different scenarios. For instance, they performed well for the classification of histological images of Barrett's cancer [16]. Our previous work [17], VGPMIL combined GPs with an attention-based CNN to address ICH diagnosis in the MIL setting. We proved that GPs outperformed the attention mechanism of CNNs for the ICH problem and set a new state-of-the-art for ICH diagnosis using only scan labels for training. To the best of our knowledge, this was the first time that GPs have been applied to the ICH diagnosis problem.

Although Gaussian Processes have not been widely used for ICH yet, they have achieved promising results on many other classification tasks [18], such as non-parametric and probabilistic models, which are capable of dealing with uncertainty in modeling and prediction [19]. Prior information can be included in the kernel function acting as a regularizer. Thus, they are not prone to overfitting. The flexibility, expressiveness, and robustness to overfitting of GPs make them suitable for a wide range of problems, especially, when only limited data is available. For this reason, they are promising for medical applications. In spite of all the benefits previously mentioned, GPs suffer from an important drawback. Commonly, they are used with stationary kernels. These kernels work well in many scenarios but they are not able to capture complex patterns, e.g., a function that is flat in one region and varies rapidly in another. Moreover, high parametrized kernels, which represent richer functions using shallow GPs, are expensive to train so approximate methods may be at risk of overfitting [20]. To overcome this limitation, DGPs have been introduced [21]. They are hierarchical extensions of GPs enabling to model more complex functions while retaining all the benefits of shallow GPs. DGPs can learn a representation hierarchy non-parametrically with very few hyperparameters [20]. DGPs have been used in medical imaging problems, such as histology, with sound results [22] against GPs and DL methods. So far the existing DGP-approaches focused on fully supervised training mostly for regression [21, 20], classification [21, 20, 22], or special cases like multi-view representation learning [23], a learning paradigm where multiple data sources with different data formats are taken into account. To the best of our knowledge, there is no existing DGP-model for the MIL setting with only bag labels available.

This work aims to extend our previous conference paper [17], which uses an attention-based MIL combined with GPs for ICH detection. We overcome the limitation of the originally applied shallow GP, which is only capable of modeling functions with limited complexity. Therefore, instead of using GPs, we propose a novel MIL method based on DGPs called DGPMIL. The new DGPMIL is more flexible than VGPMIL and improves the performance of the classifier. In this work, we also use the attention-based CNN proposed in [17] to extract the features, but this time, the hierarchical structure of DGPs enables us to capture richer patterns. In addition, the inducing locations of DGPMIL are optimized per layer in contrast to VGPMIL, the model used in [17], where they were fixed after a k-means estimation. The main contributions are:

- We introduce DGPMIL, a novel probabilistic model based on DGPs for MIL classification. To the best of our knowledge, DGPs have never been proposed before for MIL in any domain. We outline the detailed theoretical derivation and make the implementation of the model publicly available at `https://github.com/wizmik12/DGPMIL`. It is based on GPytorch, a framework for GPs on top of Pytorch, and can leverage GPU computation for fast inference.
- We study the behavior of this new MIL approach in a controlled experiment using the MNIST database. This experiment shows how the greater expressiveness of deep GPs achieves better results than shallow GPs in MIL.
- Finally, we apply the DGPMIL model combined with an attention CNN to ICH detection with labels at the scan level. These experiments demonstrate the suitability of this method to medical imaging. We report competitive or superior results to current state-of-the-art methods. Remarkably, the precision obtained at detecting ICH is notably better than previous approaches for this problem.

The rest of the paper is organized as follows. Section 2 describes the proposed model. We explain the feature extraction process using an attention-based CNN and also describe DGPMIL. Section 3 validates the method. We first create a synthetic MIL problem of digit classification to show the behavior of DGPMIL and then we perform a comprehensive validation for ICH detection on CT scans. Section 4 analyzes the main findings of the reported results and section 5 concludes our work.

2. Methods

2.1. Problem formulation

Mathematically, we model the ICH detection as a MIL problem. We denote the set of all CT slices as $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_N\}$ and the true (unobserved) slice labels as $\mathbf{Y} = \{y_1, y_2, .., y_N\}$ with $y_i \in \{0, 1\}$, where the class label 1 is assigned when the slice or scan is ICH positive and otherwise 0 if no ICH is present, and N is the total number of slices in a given bag. Note that N can vary depending on the bag. In the context of MIL, these slices are called *instances* and a complete scan (consisting of multiple slices) *bags*. The bags are non-overlapping, such that each index i of an instance can be only assigned to one bag b. We denote the instances of one bag as $\mathbf{X}_b = \{x_i | i \in \text{bag } b\}$ and corresponding instance labels as $\mathbf{Y}_b = \{y_i | i \in \text{bag } b\}$. In the MIL assumption, the instance labels remain unobserved and only the bag label T_b is known. When a CT scan

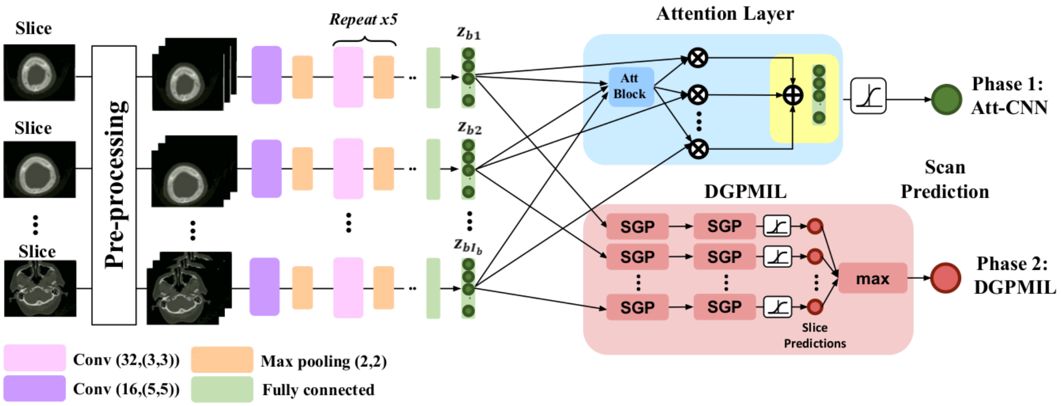


Figure 1: The proposed architecture for the ICH detection with scan labels. In phase 1 the feature extractor is trained using an attention module for bag level predictions (Att-CNN). In phase 2 the weights of the feature extractor are frozen and DGPMIL is trained to predict slice and scan level labels. We depict only a two-layer DGPMIL here although in the experiments we use a varying amount of layers to find the optimal configuration.

is diagnosed as ICH positive, at least one slice must contain the pattern of hemorrhage while a negative scan contains only negative slices, in other words,

$$T_b = max\{y_i | i \in \text{bag } b\}. \tag{1}$$

2.2. Overview of the model

To solve the MIL problem just defined, our model is trained in two phases, described in Figure 1. First, we train a convolutional neural network (CNN) that serves as a feature extractor in combination with an attention mechanism (Phase 1). The purpose of this phase is to build a feature extractor that is able to obtain expressive features from the slices. Although this phase 1 model (Att-CNN) is also able to predict ICH on CT scan level we disregard the attention layer after the first phase because we can experimentally prove that our DGPMIL model shows a stronger classification performance using the obtained features (see 3.4). The second phase consists of the classification using the extracted model features. In [17], the second phase was performed using VGPMIL. Notice that this shallow model could be too simple for the extracted features. In this work, we propose for the first time Deep Gaussian Processes for Multiple Instance Learning (DGPMIL). We describe the modeling and the inference with all derivations. The emphasis of this work lies on the training of the DGPMIL model (in phase 2) that provides the final slice and scan level predictions. We prove that DGPs take advantage of the complex patterns of the extracted features.

In the following subsections, we will briefly explain how the feature extraction is performed in our experiments.

2.3. Feature Extraction

This subsection provides a brief introduction to the attention mechanism with CNNs to extract brain CT features at slice-level, as shown in Fig.1. Assume a CNN model $\mathbf{F}_{cnn}$ is used to extract high dimensional features ξ_i for each instance x_i, such that $\xi_i = \mathbf{F}_{cnn}(x_i), \forall i = 1, 2, .., N$. Note that the same network is applied to each instance and the weights are shared. $\mathbf{F}_{cnn}$ consists of six convolutional layers, each followed by a max pooling layer. The convolutional layers aim to extract discriminative features from each instance and the max pooling layers are used to reduce the feature dimensions. Moreover, a flatten layer and a fully connected layer are followed by to control the size of feature vectors $\xi_i \in R^{M\times 1}$ fed to the attention layer and the DGPMIL model in Phase 2.

An attention layer $\mathbf{L}_{att}$ is applied after $\mathbf{F}_{cnn}$ to estimate an attention weight α_i, corresponding to each unique feature vector ξ_i. The attention weights are used to calculate a weighted sum of feature vectors for the final, bag-level classification. Let $\Xi_b = \{\xi_i | i \in \text{bag } b\}$ be the set of all feature vectors in a bag b and $\{\alpha_i | i \in \text{bag } b\}$ be the attention weights for feature vectors Ξ_b, such that $\mathbf{L}_{att}$ is defined as

$$\mathbf{L}_{att}(\Xi_b) = \sum_{i\in b} \alpha_i \xi_i, \tag{2}$$

where

$$\alpha_i = \frac{exp\{w^\top \tanh(V\xi_i)\}}{\sum_{j\in b} exp\{w^\top \tanh(V\xi_j)\}}, \tag{3}$$

$w \in R^{L\times 1}$ and $V \in R^{L\times M}$ are trainable parameters that accommodate different instance numbers of a bag. The hyperparameter L is one dimension of weight matrices w and V which defines the number of trainable parameters of the attention mechanism (and is invariant to the bag size). We set $L = 50$ following the existing literature [13]. M equals the dimension of the feature vectors, and we report the experiments for $M = 8, 32$, and 128, see section 3.4.1. The sum of all α_i in one bag is 1. The non-linearity $\tanh(\cdot)$ aims to preserve both positive and negative values during the gradient flow.

Next, the weighted sum of the feature vectors $\mathbf{L}_{att}(\Xi_b)$ is fed to a classifier $\mathbf{F}_c$, which is made up of a fully connected layer with a sigmoid activation function, to predict the scan labels, such that

$$p(T_b|X_b) = \mathbf{F}_c(\mathbf{L}_{att}(\Xi_b)) = \mathbf{F}_c(\mathbf{L}_{att}(\mathbf{F}_{cnn}(X_b))). \tag{4}$$

The feature extractor $\mathbf{F}_{cnn}$, attention layers $\mathbf{L}_{att}$ and classifier $\mathbf{F}_c$ are trained end-to-end using the basic binary cross-entropy, CE, until it converges. The loss $\mathcal{L}$ is defined as:

$$\mathcal{L} = \sum_b CE(T_b, p(T_b|X_b)). \tag{5}$$

This whole attention CNN process is denoted as Att-CNN. For more details about the attention mechanism for MIL, we refer to [13]. Previous studies show that the labels at the instance level can be inferred from the attention weights [13, 24]. The closer to 1, the more important role that specific instance contributes to the bag prediction. Therefore, in terms of this study, if a scan is predicted as normal, all slices will be considered normal. If a scan is predicted as the ICH, the slices with min-max normalized attention weights above 0.5 will be predicted as the ICH. By doing this, we are able to have weakly predicted labels at slice-level to facilitate radiologists with their diagnosis and localization. In the next section, we describe the DGPMIL model for the given problem. In what follows, to be consistent with the GP literature, we replace Ξ_b and ξ_i by $\mathbf{X}_b$ and $\mathbf{x}_i$ as the extracted feature vectors serve as an input for the final DGP classification.

2.4. Deep Gaussian Processes for Multiple Instance Learning (DGPMIL)

Here, we introduce the novel DGP model to solve the MIL problem for binary classification. We outline the basic theory of GPs and DGPS in the Appendix Appendix A and refer the reader to [19, 25, 21] for further theoretical background. Note that in contrast to previous DGP-based methods, our proposed model trains with only the bag labels T_b while the instance labels y_b are unknown, as described in subsection 2.1.. For the observation model, we follow the approach used for *Variational Gaussian Process Multiple Instance Learning* [16]. There, the authors parametrize the bag label likelihood using

$$\mathrm{p}(T_b|\mathbf{Y}_b) = \frac{H^{G_b}}{H+1}, \tag{6}$$

where $G_b := T_b \max(\mathbf{y}_b) + (1 - T_b)(1 - \max(\mathbf{y}_b))$. In this equation, H is a positive constant. Notice that this likelihood is a noisy version of the MIL assumption presented in section 2.1 and it becomes exact when H approaches infinity. The constant H controls the probability of the bag being positive considering that there is at least one positive instance. Assuming independence across bags produces the factorization

$$\mathrm{p}(\mathbf{T}|\mathbf{Y}) = \prod_{b=1}^{B} \frac{H^{G_b}}{H+1}, \tag{7}$$

where $\mathbf{T}$ refers to the variable which groups together all the bag labels.

We predict the instance label $\mathbf{y}$ by modeling a latent function $\mathbf{F}^L$ using a DGP with L layers.

Combining the Deep Gaussian Process model and the bag observation model we obtain the full probabilistic model

$$\begin{aligned}\mathrm{p}(\mathbf{Y}, \mathbf{T}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = \mathrm{p}(\mathbf{T}|\mathbf{y}) \cdot \mathrm{p}(\mathbf{y}|F^L)& \\ \prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1}),&\end{aligned} \tag{8}$$

where the dependency on the observed features $\mathbf{X}$ and the hyperparameters Θ have been omitted for simplicity.

2.5. DGPMIL inference

In this subsection, we describe the inference for our DGPMIL model. Additional details are provided in Appendix B. Our goal is to approximate the intractable posterior distribution $\mathrm{p}(\mathbf{Y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L|\mathbf{T}, \Theta)$ with an approximate distribution $\mathrm{q}(\mathbf{Y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L)$. Specifically, we perform doubly stochastic inference for DGPs [20]. We convert the inference problem into an optimization one by maximizing the Evidence Lower Bound (ELBO), defined by

$$\mathrm{ELBO(q)} = \int \mathrm{q}(\mathbf{y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) \log \frac{\mathrm{p}(\mathbf{y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L|\mathbf{T}, \Theta)}{\mathrm{q}(\mathbf{y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L)} \quad \mathrm{d}\mathbf{y}\mathrm{d}\{\mathbf{F}^l, \mathbf{U}\}_{l=1}^L. \tag{9}$$

.

In this work, we use the mean-field approximation, i.e., q factorizes across as follows:

$$\mathrm{q}(\mathbf{Z}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = \mathrm{q}(\mathbf{Y}) \times \mathrm{q}(\{\mathbf{F}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L, \Theta) \times \mathrm{q}(\{\mathbf{U}^l\}_{l=1}^L), \tag{10}$$

with the following parametric form for each factor:

$$\mathrm{q}(\mathbf{Y}) = \prod_{n=1}^{N} \mathrm{q}(y_n) = \prod_{n=1}^{N} q_n^{y_n}(1-q_n)^{1-y_n}, \tag{11}$$

$$\mathrm{q}(\{\mathbf{F}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L,\Theta) = \mathrm{p}(\{\mathbf{F}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L,\Theta), \tag{12}$$

$$\mathrm{q}(\{\mathbf{U}^l\}_{l=1}^L) = \prod_{l=1}^{L} \mathrm{q}(\mathbf{U}^l) = \prod_{l=1}^{L} \mathcal{N}(\mathbf{U}^l|\mathbf{m}^l,\mathbf{S}^l). \tag{13}$$

The proposed posterior on the instance labels $\mathbf{Y}$ factorizes across the instances and we denote by q_n the probability of the n-th instance to belong to the positive class and by $\mathbf{q}_{b-n}$ all other instance probabilities in the same bag. The prior conditional $\mathbf{F}|\mathbf{U}$ does not introduce any new variational parameter. The proposed posterior distribution on $\mathbf{U}^l$ factorizes across the layers and is given by a Gaussian distribution. In summary, the variational parameters $\mathbf{V}$ to be estimated are $\{q_n\}_{n=1}^N$ and $\{\mathbf{m}^l,\mathbf{S}^l\}_{l=1}^L$.

Finally, we obtain $\mathbf{V}$, Θ, and $\{\mathbf{Z}^l\}_{l=1}^L$ by maximizing the ELBO. The ELBO can be written explicitly as

$$\begin{aligned}
&\mathrm{ELBO}(\mathbf{V},\Theta,\{Z^{l-1}\}_{l=1}^L) = \\
&\mathbb{E}_{\mathrm{q}(\mathbf{Y})\mathrm{p}(\{\mathbf{f}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L)\mathrm{q}(\{\mathbf{U}\}_{l=1}^L)}\left[\log\frac{\mathrm{p}(\{\mathbf{U}\}_{l=1}^L)\cancel{\mathrm{p}(\{\mathbf{F}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L)}\mathrm{p}(\mathbf{Y}|\mathbf{f}^L)\mathrm{p}(\mathbf{T}|\mathbf{Y})}{\mathrm{q}(\{\mathbf{U}\}_{l=1}^L)\cancel{\mathrm{p}(\{\mathbf{F}^l\}_{l=1}^L|\{\mathbf{U}^l\}_{l=1}^L)}\mathrm{q}(\mathbf{Y})}\right] \\
&= \mathbb{E}_{\mathrm{q}(\mathbf{Y})\mathrm{p}(\mathbf{F}^L|\mathbf{U}^L)\mathrm{q}(\mathbf{U}^L)}\left[\log\mathrm{p}(\mathbf{Y}|\mathbf{f}^L)\right] + \mathbb{E}_{\mathrm{q}(\mathbf{Y})}\left[\log\mathrm{p}(\mathbf{T}|\mathbf{Y})\right] - \mathbb{E}_{\mathrm{q}(\mathbf{Y})}\left[\log\mathrm{q}(\mathbf{Y})\right] \\
&+ \mathbb{E}_{\mathrm{q}(\{\mathbf{U}^l\}_{l=1}^L)}\left[\log\frac{\mathrm{p}(\{\mathbf{U}^l\}_{l=1}^L)}{\mathrm{q}(\{\mathbf{U}^l\}_{l=1}^L)}\right].
\end{aligned} \tag{14}$$

Notice that the term $\mathbb{E}_{\mathrm{q}(\mathbf{Y})}\left[\log\mathrm{p}(\mathbf{T}|\mathbf{Y})\right]$ is not differentiable since it involves the max function. This fact prevents us from optimizing the ELBO using gradient descent. To overcome this limitation, we iteratively update first $\mathrm{q}(\mathbf{Y})$ and then the DGP parameters. Since we are using the mean-field approximation, following the approach of [16], we can compute the optimal distribution of $\mathrm{q}(\mathbf{Y})$ with the other distributions fixed [26]. The optimal update for $\mathrm{q}(\mathbf{y})$ is given by (see Appendix B.1),

$$\mathrm{q}_n \leftarrow \sigma\left(\mathbb{E}_{\mathrm{q}(f_n^L)}\left[f_n^L\right] + \log H\cdot(2T_b + \max\mathbf{q}_{b-n} - 2T_b\max\mathbf{q}_{b-n} - 1)\right). \tag{15}$$

Using the approximation $\mathbb{E}[\max\{y_i\}] \approx \max\{\mathbb{E}[y_i]\}$ as in [16], the ELBO can be approximated by (see Appendix B.2)

$$\begin{aligned}
\mathrm{ELBO} \approx &\sum_{n=1}^{N} q_n\mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log\mathrm{p}(y_n=1|f_n^L)\right] + (1-q_n)\mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log\mathrm{p}(y_n=0|f_n^L)\right] \\
&+ \log H\sum_{b=1}^{B}(2T_b\max\mathbf{q}_b - \max\mathbf{q}_b) \\
&- \sum_{n=1}^{N} q_n\log q_n + (1-q_n)\log(1-q_n) - \sum_{l=1}^{L}\mathrm{KL}\left(\mathrm{q}(\mathbf{U}^l)||\mathrm{p}(\mathbf{U}^l)\right) \\
&+ \text{const.}
\end{aligned} \tag{16}$$

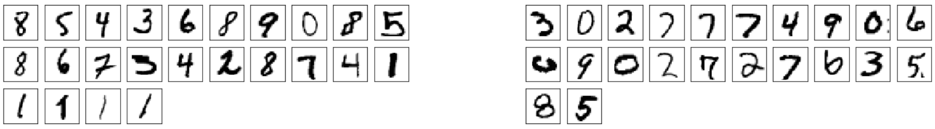
Now, with q_n fixed, we can optimize the ELBO in eq. (16) to obtain the optimal distribution for q($\{\mathbf{U}^l\}_{l=1}^L$), the kernel hyperparameters Θ and the inducing locations $\{\mathbf{Z}^l\}_{l=1}^L$ by using gradient descent (see Appendix B.3). Then, we can compute the variational parameters q_n with the update in eq. (15) where the other parameters are fixed. As we commented before, this optimization is performed iteratively.

3. Experiments

This section provides an empirical validation of the proposed DGPMIL model. We carry out two different experiments. First, we create a synthetic toy example based on the popular MNIST dataset to show the behavior of DGPMIL against VGPMIL [16] in a controlled environment. Then, we use the features extracted by the attention-based CNN presented in section 2.3 with both VGPMIL and DGPMIL for clinical ICH detection. We show the capacity of DGPMIL against the previous VGPMIL [17] and other state-of-the art methods in this problem.

3.1. Toy example: MNIST

To see the behavior of the novel DGPMIL, we analyze a synthetic MIL problem using the MNIST dataset. MNIST has 60,000 training samples and 10,000 test samples and each instance is composed of a 784-dimensional feature vector. We want to compare a shallow GP model with deep GP models to evaluate their capacity to handle high-dimensional, complex feature distributions. Since it is a controlled experiment, we carry out a comprehensive analysis to highlight its main properties. The availability of instance labels allows us to assess the model at both instance and bag levels.



(a) A positive bag

(b) A negative bag

Figure 2: Examples of bags in the training set for the MNIST experiment.

In our MNIST synthetic problem, bags contain images of numbers between 0 and 9. The goal is to decide whether the bag contains at least one image of a one and, if possible, to localize it (them) in the bag. Each positive bag contains 1 to 10 positive (images of ones), and 10 to 30 negative (other numbers) instances. Negative bags contain only negative, specifically 10 to 30 negative instances. In total, we obtain 1416 negative and 1333 positive bags for training. Figure 2 shows two examples of bags in the training set. The 10,000 samples of the test set are distributed in 229 negative and 231 positive bags. We compare DGPMIL and VGPMIL models in our experiment. For the Deep Gaussian Process model, we compare the performance with 2, 3, and 4 GP layers. The dimension of the latent space of the hidden layers is set to 7 for every layer, 200 inducing points are used for each model per layer. We compute the accuracy in the test set at both instance and bag level. To assess the confidence of the methods, we also compute the log loss over the test set.

Table 1: Results in the MNIST of Multiple Instance Methods based on Gaussian Processes using the first 30 principal components after using PCA. VGPMIL is the shallow Variational GP while DGPMIL is the deep version with 2, 3, and 4 GP layers. We assess the classification performance both at the instance and bag levels.

	Instance level		Bag level	
	Accuracy	Log Loss	Accuracy	Log Loss
VGPMIL	0.9767	0.6006	0.8496	0.4016
DGPMIL2	0.9896	0.2672	0.9586	0.2859
DGPMIL3	**0.9913**	0.2638	**0.9760**	0.2531
DGPMIL4	0.9909	**0.2602**	**0.9760**	**0.2517**

3.1.1. Dimensionality reduction with PCA

Shallow methods are not good at dealing with high-dimensional complex data. This is one of the main reasons for the advent of hierarchical methods. For a fair comparison, we first reduce the dimensionality of data with Principal Component Analysis (PCA) and keep the first 30 principal components for each digit image. In the next experiment, we apply VGPMIL and DGPMIL to the raw MNIST. By doing this, we can analyze and discern the relevance of deep methods in both low and high-dimensional contexts.

Table 1 shows the comparison between VGPMIL and DGPMIL for this experiment. VGPMIL achieves a good instance classification with a value of 0.9767 in accuracy but lower for bag classification with 0.8496. In contrast, DGPMIL shows a good performance for both, instance and bag classification. For example, DGPMIL3 obtains 0.9913 at the instance level and 0.9760 at the bag level. In general, DGPMIL outperforms VGPMIL at the bag level. Regarding the log loss, VGPMIL performs poorly at the instance level which indicates that the high uncertainty lowers the overall bag classification. Although we reduced the complexity of this problem by the PCA preprocessing, we observe that the deeper GP models achieve significantly better performance on the bag level.

3.1.2. Raw MNIST data

Table 2 shows the comparison between VGPMIL and DGPMIL on the raw MNIST data. Due to the high-dimensionality of this dataset and the simplicity of the classifier, VGPMIL performs poorly. This table shows that it predicts always the positive class at the instance and bag level. That is the reason why it reaches a value of 0.11 in accuracy for instance evaluation, while reaches a value of 0.49 in accuracy for bag evaluation. In contrast, deep models are able to process this high-dimensional data and provide accurate predictions. We can see that the best instance classifier is the deepest model DGPMIL4 with an accuracy of 0.9932 and log loss of 0.2533, followed by DGPMIL3, which achieves the best result at bag level with an 0.9717 accuracy and of 0.2519 log loss.

3.2. CT scan

So far, we have seen the behavior of DGPMIL in a controlled experiment. It shows a satisfying performance against its shallow version, i.e., VGPMIL. Now, we study the performance of an attention-based CNN combined with GP-based methods in a real-world problem. We tackle the problem of detecting ICH on brain CTs in a MIL setting. We analyze the advantages produced by using a DGP classifier on the top of the CNN instead of a shallow GP, which was presented in [17]. We consider a full scan as a bag

Table 2: Results in the MNIST of Multiple Instance Methods based on Gaussian Processes using the 784-dimensional feature vector. VGPMIL is the shallow Variational GP while DGPMIL is the deep version with 2, 3, and 4 GP layers. We assess the classification performance both at the instance and bag level.

	Instance level		Bag level	
	Accuracy	Log Loss	Accuracy	Log Loss
VGPMIL	0.1135	0.6931	0.4978	0.6931
DGPMIL2	0.9857	0.2729	0.9304	0.3113
DGPMIL3	0.9930	0.2655	**0.9717**	**0.2519**
DGPMIL4	**0.9932**	**0.2533**	0.9652	0.2598

and each slice in a scan as an instance. Generally, different scans contain a different number of slices. So in this case, the number of instances in bags varies.

3.2.1. Data Preprocessing

The used dataset was published by the Radiological Society of North America (RSNA) [1] in 2019. This study includes a total of 39750 slices acquired from 1150 patients, which are further split into 1000 subjects for training and validation, and the rest 150 subjects for testing. Specifically, the training dataset includes 589 normal scans (i.e., negative cases) and 411 scans with ICH (i.e., positive cases) and the testing dataset includes 78 normal scans and 72 ICH scans. The number of slices in each scan ranges from 24 to 57 in size of 512 × 512. At slice-level, the training dataset includes 29520 negative slices and 4976 positive slices and the testing dataset includes 806 positive slices and 4448 negative slices.

The CQ500 dataset provided by various centers in New Delhi, India [27] is used as an external test set in this study to show the generalization of our proposed model trained on RSNA. It includes the ground truth only at scan-level, including 285 normal scans and 205 ICH scans. The number of slices in each scan varies from 16 to 128.

In both datasets, in order to mimic the way radiologists often adjust different window centers (C) and widths (W) when diagnosing a brain scan, each slice is passed through three window settings to enhance the different display of the brain [W:80, C:40], blood [W:200, C:80] and soft tissue [W:380, C:40]. The windowing images from each slice are stacked together as three image channels and the intensities are normalized to [0,1] before being fed into the CNNs.

3.3. Implementation details

The model is first trained with an attention CNN with the ground truth at scan-level where the estimated attention weights will indicate the probability of that slice being positive. Then, the features at slice level can be extracted from the fully connected layers. Finally, these extracted features are fed into VGPMIL and DGPMIL.

The attention CNN is trained from scratch (without pre-trained weights) and the whole training procedure costs an average of 4.5 hours. The number of training epochs is 100 and the batch size is 16 per step. The Adam optimizer [28] is used with an initial learning rate of 5×10^{-4}. The experiment is run five times independently and both the

[1] https://www.kaggle.com/c/rsna-intracranial-hemorrhage-detection/)

training and testing processes are performed on a single GPU (Nvidia GeForce RTX 2070 Super) using Tensorflow 2.0 and Python 3.7.

In this experiment, we compare the performance of GP-based methods and deep neural networks. We use the shallow VGPMIL and three different values of depth for DGPs: 2-layer (DGPMIL2), 3-layer (DGPMIL3) and 4-layer (DGPMIL4) models. The training of DGP models is performed with Adam optimizer, 512 mini-batch size and 30 epochs. Furthermore, the dimension of the latent space has been set to 3 for hidden layers. After several tries, we see empirically that a small latent space benefit and accelerate convergence. The learning rate is set to 0.001. While for VGPMIL we use the published implementation in NumPy of [16], DGPMIL is implemented using GPyTorch 1.3.1 , which is a software for GPs based on PyTorch. The used version of PyTorch is 1.7.1.

3.4. Results

In this section, we report the results for ICH detection. First, we study the impact of the hyperparameters to the model's performance. Then, we test the model in the RSNA and the external CQ500 databases. Finally, we compare the performance of the DGPMIL to other state-of-the-art classifiers in ICH.

To measure the performance of the different variants of DGPMIL and compare to other state-of-the-art methods, we mainly use three important metrics: F1 score, Area Under the Curve of the receiver operating characteristic (ROC-AUC) and the precision-recall (PR-AUC) curve. The F1 score measures the performance based on precision and recall and is a common machine learning metric that is also suitable for class-imbalanced scenarios. The ROC plots the true positive rate against the false positive rate for different confidence thresholds of the model. Here, a good model can obtain a high true positive rate while maintaining a low false positive rate. The precision-recall curve plots precision against recall for different confidence thresholds. All three metrics have a range between 0 and 1 and the higher the value, the better.

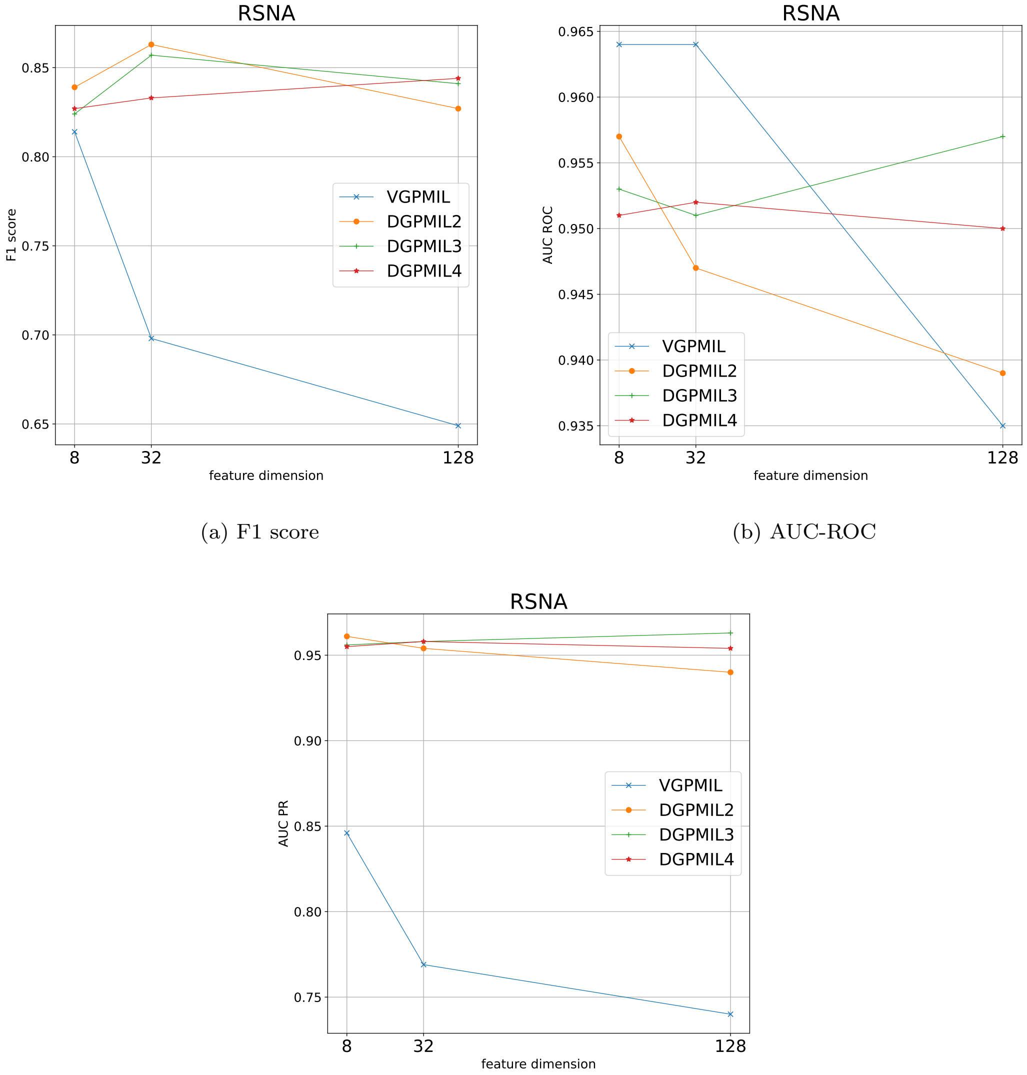
3.4.1. Ablation Studies

This subsection studies the characteristics of the DGPMIL model and its hyperparameters. We conducted an ablation study. Specifically, we report the impact of the number of feature dimensions, the number of DGPMIL layers, the number of inducing points, and the dimensionality of the latent space on GPs' performance.

We start by analyzing the effect of different feature space dimensions M of the vectors ξ_i that enter the DGPMIL model and the number of GP layers, i.e., the depth of the proposed model. We compare the shallow VGPMIL to the DGPMIL models with 2, 3, and 4 layers for 8, 32, and 128-dimensional input features. We measured the performance at the scan (bag) level. During these experiments, we fixed the number of inducing points to 200 and the latent space dimensions to 3. See below for an analysis of these hyperparameters.

Figure 3 shows the results for the RSNA dataset, while Figure 4 shows the results for the CQ500 dataset. Both figures report F1 score, AUC-ROC, and AUC-PR metrics. As we can observe in all figures, the shallow VGPMIL model could not achieve satisfying results for higher feature dimensions. We measured some significant performance drops, e.g., the AUC-ROC for the CQ500 dataset (Figure 4b) drops by 5% for 32 feature dimensions and 10% for 128 feature dimensions. The DGPMIL models show more robust performance in all three metrics, and even with 128-dimensional feature vectors, they

achieve satisfying results. Within the different DGPMIL models, higher feature dimensions seem to harm the DGPMIL2 model the most, as the performance drops are larger than for the DGPMIL3 and DGPMIL4 models for all AUC metrics (Figures 3b, 3c, 4b, 4c). Regarding the F1 score, we can even see improved performance when using more feature dimensions. The DGPMIL3 and DGPMIL4 models both show a better F1 score when using 128 dimensions in comparison to 8 on both datasets (see Figures 3a and 4a). Overall we observed that DPGMIL can learn useful models from feature vectors of higher dimensions while the shallow VGPMIL can not. In section 4, we further discuss this interesting relationship between feature dimensions and GP layers.
In the final experiments, we stick to DGPMIL2 with 8 feature dimensions because this setting still achieves the best results on both datasets in terms of AUC-ROC and AUC-PR.

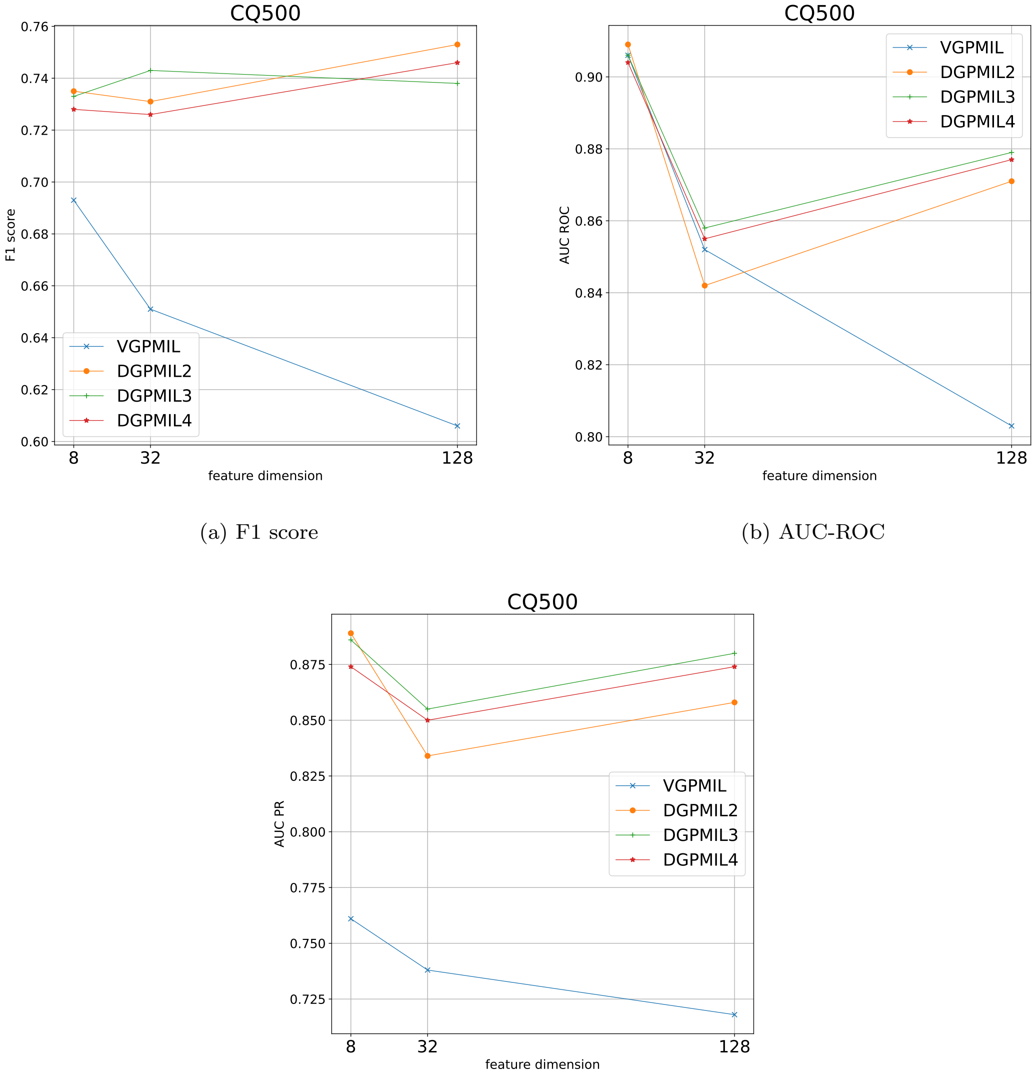


(a) F1 score

(b) AUC-ROC

(c) AUC-PR

Figure 3: RSNA dataset: F1 score, AUC-ROC and AUC-PR for GP and DGP models using different input feature dimensions.



(a) F1 score

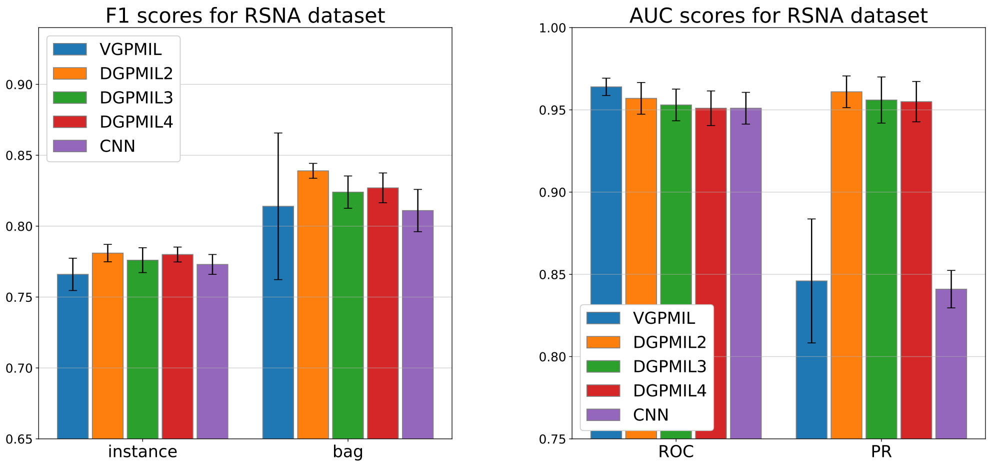
(b) AUC-ROC

(c) AUC-PR

Figure 4: CQ500 dataset: F1 score, AUC-ROC and AUC-PR for GP and DGP models using different input feature dimensions.

Next, we studied the effect of varying the number of inducing points while leaving the feature dimensions fixed at 8 and latent space dimensions at 3. As reported in Table C.1, we observed a robust performance across different numbers of inducing points. 200 inducing points show the best F1 scores for both datasets and the best AUC ROC for the RSNA dataset, we use this setting for the following experiments. Further increasing the number of inducing points did not provide any significant improvement and led to higher computational costs. Similarly, we conducted experiments to prove that the relatively small number of GP's latent space dimensions of $D = 3$ is enough. Table C.2 shows that the performance of the model with 3 and 10 latent dimensions is comparable, while 50 dimensions lead to a model that can not converge anymore.

In summary, we observed that the DGPMIL model is not very sensitive to the analyzed hyperparameters. In these experiments we made an interesting observation: for



(a) F1 score for RSNA dataset

(b) AUC values for RSNA dataset

Figure 5: RSNA dataset with 8-dimensional features: F1 score and AUC values with 0.95 confidence interval.

higher-dimensional feature vectors, more GP layers should be used because the shallow VGPMIL model is not able to obtain good results. This finding is further discussed in section 5. For the final results, we used 8 dimensional feature vectors, 200 inducing points, and 3 dimensions in the GP latent space. For the number of GP layers, all model variants are included in the experiments under the names DGPMIL2, DGPMIL3, and DGPMIL4.

3.4.2. Results for the RSNA dataset

Table 3 shows the results of testing with the RSNA dataset for 8-dimensional features. For this test set, although models are trained with only the scan labels, we have both slice and scan labels to evaluate the model performance. We reported the performances of the Attention-CNN model, VGPMIL, and DGPMIL with a different number of layers. Mean-aggregation of the feature vectors was previously analyzed for this problem [17] and can be considered a simple baseline with a bag-level ROC-AUC of 0.768. Regarding our analyzed models, the CNN model obtains the worst results and coupling the feature vectors to GPs improves the performance considerably. For most of the metrics at slice and scan levels, we see that DGPMIL2 shows the best performance.

Figure 5 shows F1 score and AUC values with 0.95 confidence interval. We can see that VGPMIL has a high variance for the F1 score and AUC-PR at the bag label while DGPMIL obtains good results with tight confidence intervals. This shows that DGPMIL is more robust. Furthermore, the non-overlapping intervals of DGPMIL against its competitors at the AUC-PR show visually the statistically significant improvement of DGPMIL thanks to the better precision.

Some examples of DGPMIL predictions for the RSNA dataset can be found in Figure 6. Furthermore, we include some misclassified slices in Figure 7. Fig. 7a and b are false negatives with prediction probabilities of 0.23 and 0.16. We found that they are both the only positive slice in their own scans, so the model is more difficult to detect those small and mild types of hemorrhage. Fig. 7c is a false positive slice predicted from an ICH scan with probability of 0.60. It is adjacent to a positive slice, so it might be predicted as positive because some bleeding can still be found in this slice. Fig. 7d is a false positive slice predicted from a normal scan with probability of 0.59. In this case, although the probability is low and close to 0.5, a false positive slice will lead to an overall positive scan prediction. Therefore, in order to handle all these challenges, for future work we propose to not treat the instances independently but focus more on the correlations among the instances, i.e., the sequence of the slices in a scan.

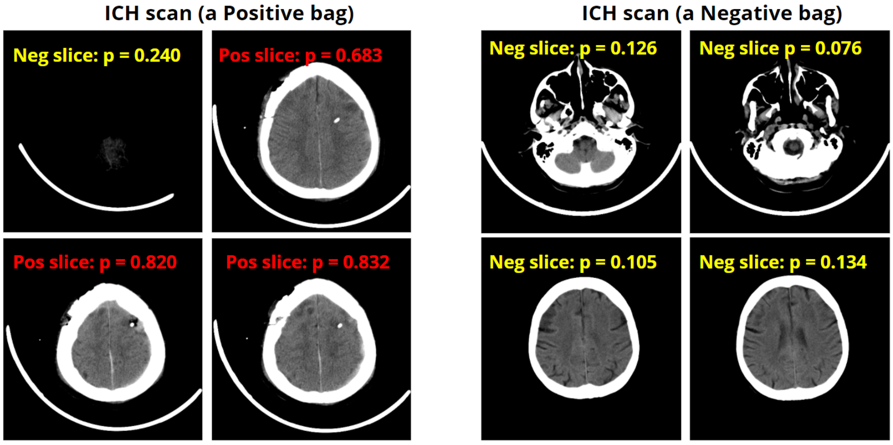


Figure 6: Examples of two bags with DGPMIL predictions at bag-level and at instance-level. Left: an ICH scan with a bag prediction of 0.834; Right: a normal scan with a bag prediction of 0.217. Probability $p \geq 0.5$ denotes an ICH prediction is positive and $p < 0.5$ denotes a negative ICH prediction. The model is trained at bag-level but it is able to provide individual instance label correctly as the p values indicate.

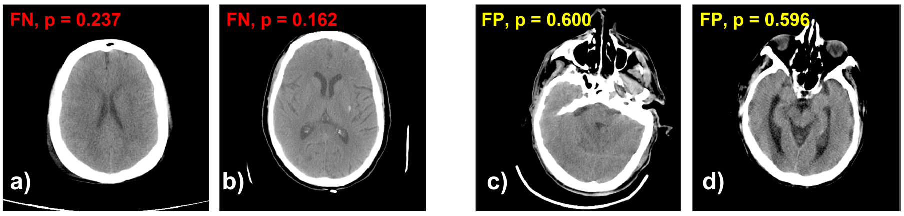


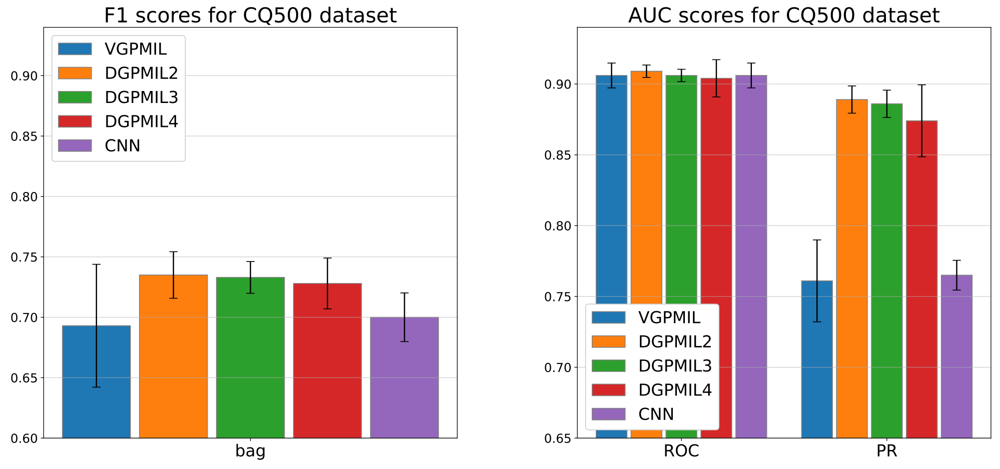
Figure 7: Examples of False Negatives (FN) and False Positives (FP) with DGPMIL predictions at the instance level.(a,b) False Negatives; (c) a False Positive from a positive bag; (d) a False Positive from a negative bag. Probability $p \geq 0.5$ denotes an ICH prediction is positive and p <0.5 denotes a negative ICH prediction.

3.4.3. Results for the external database CQ500

Table 4 shows the results of our trained model (on RSNA) tested with the CQ500 dataset for 8-dimensional features. For this test set, we only have scan labels. DGPMIL2 outperforms all other models in all metrics. Especially in the Cohen's Kappa value and AUC-PR we can see huge improvements in comparison to the CNN and VGPMIL model. Figure 8 shows F1 score and AUC values with 0.95 confidence interval. We can see that VGPMIL has a large variance both for the F1 score and AUC-PR metrics. Again, DGPMIL, specially DGPMIL2 and DGPMIL3, obtains a tight confidence interval even when generalizing to an external database. The non-overlapping confidence intervals show the statistical superiority of the proposed DGPMIL in AUC-PR.

Table 3: Mean results testing with the RSNA dataset for 8-dimensional features in five different runs at both slice and scan level. VGPMIL is the shallow Variational GP while DGPMIL is the deep version with 2, 3, and 4 GP layers. The CNN stands for the attention-based CNN.

Slice level metrics	VGPMIL	DGPMIL2	DGPMIL3	DGPMIL4	CNN
Accuracy	**0.938±0.003**	0.929±0.003	0.927±0.005	0.928±0.002	0.923±0.005
F1 score	0.766±0.013	**0.781±0.007**	0.776±0.01	0.780±0.006	0.773±0.008
Cohen's kappa	0.731±0.015	**0.739±0.009**	0.732±0.013	0.737±0.007	0.727±0.011
Scan level metrics	VGPMIL	DGPMIL2	DGPMIL3	DGPMIL4	CNN
Accuracy	0.780±0.089	**0.825±0.006**	0.805±0.014	0.809±0.018	0.781±0.023
F1 score	0.814±0.059	**0.839±0.006**	0.824±0.013	0.827±0.012	0.811±0.017
Cohen's kappa	0.567±0.172	**0.654±0.011**	0.614±0.029	0.622±0.035	0.569±0.045
AUC-ROC	**0.964±0.006**	0.957±0.011	0.9530±0.011	0.951±0.012	0.951±0.011
AUC-PR	0.846±0.043	**0.961±0.011**	0.956±0.016	0.955±0.014	0.841±0.013



(a) F1 score for CQ500 dataset

(b) AUC values for CQ500 dataset

Figure 8: CQ500 dataset with 8-dimensional features: F1 score and AUC values with 0.95 confidence interval.

3.4.4. State-of-the-art comparison

The performance of DGPMIL is compared with those state-of-the-art studies, as shown in Table 5. It shows that our model outperforms other models trained at scan-level with an AUC-ROC of 0.957, including basic MIL [24], 3D CNNs [9, 8, 29], and 3D autoencoder [30]. In addition, it is comparable to VGPMIL [17] with an AUC-ROC of 0.964. Note, that in this case, different scan-level approaches for ICH detection are compared that are using different datasets. Therefore we add a comparison of different models for the CQ500 dataset, where all models are tested on the same set. At the same time, this dataset serves as an external test set (as described above) because the model is trained on the RSNA dataset. DGPMIL achieves an AUC-ROC of 0.909, which performs better than the methods that are trained at the same scan-level with an AUC-ROC of 0.906 [17] and 0.83 [31]. Furthermore, the performance of DGPMIL is comparable to those trained at slice-level [27, 32], where the AUC-ROC scores ranged from 0.94-0.96.

Table 4: Mean results testing with the CQ500 dataset for 8-dimensional features in five different runs at scan level. VGPMIL is the shallow Variational GP while DGPMIL is the deep version with 2, 3, and 4 GP layers. The CNN stands for the attention-based CNN.

Scan level metrics	VGPMIL	DGPMIL2	DGPMIL3	DGPMIL4	CNN
Accuracy	0.639±0.106	**0.717±0.035**	0.713±0.023	0.701±0.041	0.655±0.043
F1 score	0.693±0.058	**0.735±0.022**	0.733±0.015	0.728±0.024	0.700±0.023
Cohen's kappa	0.335±0.171	**0.461±0.059**	0.455±0.039	0.436±0.068	0.359±0.069
AUC-ROC	0.906±0.010	**0.909±0.005**	0.906±0.005	0.904±0.015	0.906±0.010
AUC-PR	0.761±0.033	**0.889±0.011**	0.886±0.011	0.874±0.029	0.765±0.012

Table 5: Comparison of different approaches for binary ICH detection. Our results are reported as the mean of 5 independent runs.

ICH detection at scan-level with different dataset				
Source	Dataset size	Labeling type	Method	ROC AUC
Saab et al. [24]	4340 scans	Scan	MIL	0.91
Jnawali et al. [9]	40357 scans	Scan	3D CNNs	0.87
Titano et al. [8]	37236 scans	Scan	3D CNNs	0.88
Sato et al. [30]	126 scans	Scan	3D Autoencoder	0.87
Arbabshirani et al. [29]	45583 scans	Scan	3D CNNs	0.85
VGPMIL (Wu et al. [17])	1150 scans	Scan	MIL	0.964
DGPMIL2	1150 scans	Scan	MIL	0.957
Evaluation on CQ500				
Source	Dataset size	Labeling type	Method	ROC AUC
Chilamkurthy et al. [27]	490 scans	Slice	2D CNNs	0.94
Nguyen et al. [32]		Slice	2D CNN + LSTM	0.96
Monteiro et al. [31]		Scan	voxel-based CNN	0.83
VGPMIL (Wu et al. [17])		Scan	MIL	0.906
DGPMIL2		Scan	MIL	0.909

4. Discussion

In MIL problems, having a good instance classifier does not necessarily lead to a good bag classification. For the MIL setting, one misclassification of one instance leads to the wrong classification of a full bag. For this reason, well-calibrated models are desirable in MIL. The introduction of DGPMIL overcomes this problem and reaches much better classification performance at the bag level. Furthermore, it still retains a good instance performance, making it suitable for classifying new unseen or unlabeled instances.

DGPMIL achieves State-of-the-art results and generalizes better. Table 5 compares the ICH prediction results with other methods at scan-level. DGPMIL outperforms other methods based on AUC-ROC score except for VGPMIL [17], but DGPMIL performs significantly better than [17] in AUC-PR score and F1 score as previously discussed. Furthermore, we include an external database (CQ500) to check the generalization capability of our proposed models. In this real-world scenario, we are more interested in training a model on a dataset from a center and using it to predict correctly on the dataset from another center. The external evaluations on CQ500 dataset show that DGPMIL outperforms other models in Table 4, which proves the good generalization of our model. We further compare the performance of DGPMIL on CQ500 with those state-of-the-art studies in Table 5. It shows that DGPMIL outperforms other methods train with the same labeling type on the scan [17, 31] and it is comparable to other studies that training with precise slice labels [27, 32]. It is remarkable that

DGPMIL2 performs well across all different feature spaces. In addition, by selecting the number of layers, we can adjust the model to extract features with different dimensions. Since DGPMIL achieves good predictions at scan level, it is the most suitable for diagnosis on unseen scans from different centers.

DGPMIL is able to achieve good results with complex high-dimensional data. We have seen in the MNIST experiment (Section 3.1) as well as in the ablation studies of the hemorrhage classification problem (Subsection 3.4.1) that the DGPMIL model can handle complex, high-dimensional feature distributions while the shallow VGPMIL model shows significant performance drops. This can be explained by the better ability to approximate complex functions due to multiple stacked GP layers. It enables the model to transform the feature distribution in the latent space, as depicted in the explanatory example of Figure A.12, and leads to higher expressiveness. This property makes the DGPMIL especially interesting for other problems with a fixed, high number of feature dimensions where the DGPMIL model can be expected to outperform shallow models like VGPMIL by even a larger margin than in our final results with 8-dimensional features.

DGPMIL outperforms VGPMIL in a synthetic example. The first experiment is compared DGPMIL and VGPMIL models on a synthetic example using the MNIST dataset. Regarding the instance classification, the overall performance of DGPMIL is only slightly better than VGPMIL when PCA is implemented. This indicates that for a problem with low-dimensional extracted features, both shallow and deep models perform well when classifying instances. However, this is not the case for bag classifications where DGPMIL outperforms VGPMIL and it corroborates the premise of a good instance classification is not enough. The proposed DGPMIL overcomes this limitation and is more suitable for MIL problems than the previous VGPMIL. As shown in Table 2, without a previous feature extraction on MNIST dataset, VGPMIL is not able to learn a good model.

Coupling an attention-based CNN with GPs produces better results. Although CNNs are widely applied in different areas of medical images, using only a standard CNN in MIL problems is not good enough because many details in bags are hidden. For the ICH detection task, we show that the CNN predictions can be substantially improved by further utilizing the extracted features with GP models (i.e., both VGPMIL and DGPMIL), leading to better instance and bag classification results. As shown in Fig. 6, with the features extracted by an attention-based CNN, DGPMIL is able to train images at scan-level and accurately predict images at slice-level. This fact encourages the use of GP models for ICH detection without radiologists' manual annotations on each slice. Since probabilistic models quantify better the uncertainty and are therefore even more adequate for this medical diagnosis scenario than deterministic model such as standard CNNs.

DGPMIL retains a good precision. The F1 score achieved by DGPMIL is better than that obtained by the CNN and VGPMIL. Considering the AUC of the ROC and PR curves, we observe that although VGPMIL and the CNN show good AUC-ROC results, their AUC-PR results are worse, meaning that the precision scores of these models are poor compared with DGPMIL. In other words, both VGPMIL and the CNN produce many false positives, which overload the doctors with a lot of false ICH detections. DGPMIL is capable of detecting suspicious cases with a high precision, as shown in Table 3, that the AUC-PR of DGPMIL2 for RSNA dataset reaches that of 0.961.

DGPMIL performs much better at the bag level. This fact has been already reflected in the synthetic example of MNIST and has been further confirmed on a real-world CT scan experiment. Although sometimes VGPMIL achieves a good classification on CT slices, DGPMIL outperforms VGPMIL at scan level. In terms of MIL problem, misclassifying only one instance in a negative bag will ruin the classification of the full bag. This is the reason why both VGPMIL and the CNN misclassify many negative bags with false positives because they cannot handle the uncertainty quantification while DGPMIL achieves the best precision and as a consequence reaches a better diagnosis at the bag level.

Advantages and drawbacks of DGPMIL: Our approach is an attractive alternative to attention CNNs for MIL that achieves good performance by integrating a probabilistic model, Gaussian Processes. In addition, compared to other weakly supervised learning methods [8, 9], DGPMIL is easy to train as it does not have many hyperparameters or model parameters and can be used even with limited computing power. This work exploits its formulation to achieve a satisfying performance compared to previous methods for ICH detection, as shown in Table 5, at both scan-level and slice-level. Furthermore, the AUC-PR results are remarkable in comparison to other models in Table 3 and Table 4. This metric indicates that it is not prone to have many false positives, which is important for medical applications to not distract from the really severe cases. Furthermore, it is robust to overfitting and generalizes better than other methods on external testing dataset [31, 17]. However, as DGPMIL can not deal with images directly, it relies on a first step based on a CNN for feature extraction. Although this adds on extra training and parameter tuning procedures, it shows that our method can generalize well to other MIL problems [33] by just exchanging the feature extractors. Future work will focus on building an end-to-end training CNNs and GPMIL model. Another drawback of DGPMIL is that it does not take the order of the instances into account. Instances are trained independently in a bag, but the correlations existing in nearby instances may boost the performance of the model. Future work will try to implement some sequential models [32] into DGPMIL to extract the features among the order of instances.

5. Conclusions

In this work, we propose a novel model, DGPMIL, for MIL classification based on DGPs. DGPs are a hierarchical extension of the widely used GPs. Furthermore, we use DGPMIL for ICH detection on CT scans combined with the features extracted by an attention-based CNN using only scan labels. To the best of our knowledge, this is the first time DGPs have been proposed for the MIL problem and specifically for ICH detection.

The experiments show that DGPMIL can obtain good results with high-dimensional data by extracting more complex patterns in contrast to the shallow VGPMIL. For instance, DGPMIL outperforms VGPMIL in a synthetic MIL problem of classifying digits using the MNIST database. When using data with dimensionality reduction, VGPMIL performs slightly worse at the instance level compared to deep versions. However, when raw MNIST is used, VGPMIL can not learn a good model. Furthermore, DGPMIL performs notably better at the bag level, which is the final objective of the MIL problem.

We empirically validate the model in a real-world application. We detect ICH on CT scans using only scan labels. The experiment results demonstrate that combining a CNN with a GP leads to an improvement in the results. DGPMIL achieves the best performance compared to VGPMIL, the attention CNN and other state-of-the-art methods. Furthermore, it achieves a great precision value in contrast to VGPMIL and the attention CNN.

Additionally, we use a different database for assessing the generalization capability of the methods. This evaluation proves that DGPMIL generalizes better when predicting at scan level. All of these facts make DGPMIL with an attention-based CNN suitable for ICH diagnosis. Also, it can potentially be applied to many other medical-imaging problems.

References

[1] D. Kushner, Mild traumatic brain injury: Toward understanding manifestations and treatment, Archives of Internal Medicine 158 (15) (1998) 1617.

[2] J. A. Caceres, J. N. Goldstein, Intracranial hemorrhage, Emergency medicine clinics of North America 30 (3) (2012) 771–794.

[3] C. A. Taylor, Traumatic brain injury–related emergency department visits, hospitalizations, and deaths — united states, 2007 and 2013, Morbidity and Mortality Weekly Report (MMWR) Surveillance Summaries 66 (2017).

[4] W. M. Strub, J. L. Leach, T. Tomsick, A. Vagal, Overnight preliminary head CT interpretations provided by residents: Locations of misidentified intracranial hemorrhage, American Journal of Neuroradiology 28 (9) (2007) 1679–1682.

[5] W. K. Erly, W. G. Berger, E. Krupinski, J. F. Seeger, J. A. Guisto, Radiology resident evaluation of head CT scan orders in the emergency department, American Journal of Neuroradiology 23 (1) (2002) 103–107.

[6] T. D. Phong, H. N. Duong, H. T. Nguyen, N. T. Trong, V. H. Nguyen, T. Van Hoa, V. Snasel, Brain hemorrhage diagnosis by using deep learning, in: International Conference on Machine Learning and Soft Computing, 2017, pp. 34–39.

[7] J. Cho, K.-S. Park, M. Karki, E. Lee, S. Ko, J. K. Kim, D. Lee, J. Choe, J. Son, M. Kim, S. Lee, J. Lee, C. Yoon, S. Park, Improving sensitivity on identification and delineation of intracranial hemorrhage lesion using cascaded deep learning models, Journal of Digital Imaging 32 (3) (2019) 450–461.

[8] J. J. Titano, M. Badgeley, J. Schefflein, M. Pain, A. Su, M. Cai, N. Swinburne, J. Zech, J. Kim, J. Bederson, J. Mocco, B. Drayer, J. Lehar, S. Cho, A. Costa, E. K. Oermann, Automated deep-neural-network surveillance of cranial images for acute neurologic events, Nature Medicine 24 (9) (2018) 1337–1341.

[9] K. Jnawali, M. R. Arbabshirani, N. Rao, A. A. P. M.d, Deep 3d convolution neural network for CT brain hemorrhage classification, in: Medical Imaging 2018: Computer-Aided Diagnosis, Vol. 10575, International Society for Optics and Photonics, 2018, p. 105751C.

[10] M.-A. Carbonneau, V. Cheplygina, E. Granger, G. Gagnon, Multiple instance learning: A survey of problem characteristics and applications, Pattern Recognition 77 (2018) 329–353.

[11] G. Campanella, M. G. Hanna, L. Geneslaw, A. Miraflor, V. Werneck Krauss Silva, K. J. Busam, E. Brogi, V. E. Reuter, D. S. Klimstra, T. J. Fuchs, Clinical-grade computational pathology using weakly supervised deep learning on whole slide images, Nature Medicine 25 (8) (2019) 1301–1309.

[12] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7132–7141.

[13] M. Ilse, J. Tomczak, M. Welling, Attention-based deep multiple instance learning, in: International Conference on Machine Learning - ICML, 2018, pp. 2127–2136.

[14] Z. Han, B. Wei, Y. Hong, T. Li, J. Cong, X. Zhu, H. Wei, W. Zhang, Accurate Screening of COVID-19 Using Attention-Based Deep 3D Multiple Instance Learning, IEEE Transactions on Medical Imaging 39 (8) (Aug. 2020).

[15] S. Qi, C. Xu, C. Li, B. Tian, S. Xia, J. Ren, L. Yang, H. Wang, H. Yu, DR-MIL: deep represented multiple instance learning distinguishes COVID-19 from community-acquired pneumonia in CT images, Computer Methods and Programs in Biomedicine 211 (2021) 106406.

[16] M. Haußmann, F. A. Hamprecht, M. Kandemir, Variational bayesian multiple instance learning with gaussian processes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6570–6579.

[17] Y. Wu, A. Schmidt, E. Hernández-Sánchez, R. Molina, A. K. Katsaggelos, Combining attention-based multiple instance learning and gaussian processes for CT hemorrhage detection, in: Medical Image Computing and Computer Assisted Intervention – MICCAI, 2021, pp. 582–591.

[18] J. Hensman, A. G. de G. Matthews, Z. Ghahramani, Scalable variational gaussian process classification., in: Artificial Intellgince and Statistics (AISTATS), Vol. 38, 2015.

[19] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2006.

[20] H. Salimbeni, M. Deisenroth, Doubly stochastic variational inference for deep gaussian processes, in: Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 4588–4599.

[21] A. Damianou, N. Lawrence, Deep Gaussian processes, in: International Conference on Artificial Intelligence and Statistics, Vol. 31, 2013, pp. 207–215.

[22] Ángel E. Esteban, M. López-Pérez, A. Colomer, M. A. Sales, R. Molina, V. Naranjo, A new optical density granulometry-based descriptor for the classification of prostate histological images using shallow and deep gaussian processes, Computer Methods and Programs in Biomedicine 178 (2019) 303–317.

[23] S. Sun, W. Dong, Q. Liu, Multi-view representation learning with deep gaussian processes, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (12) (2021) 4453–4468.

[24] K. Saab, J. Dunnmon, R. Goldman, A. Ratner, H. Sagreiya, C. Ré, D. Rubin, Doubly weak supervision of deep learning models for head CT, in: Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Vol. 11766, 2019, pp. 811–819.

[25] E. Snelson, Z. Ghahramani, Sparse gaussian processes using pseudo-inputs, in: Advances in Neural Information Processing Systems, Vol. 18, MIT Press, 2006.

[26] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[27] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, P. Warier, Development and validation of deep learning algorithms for detection of critical findings in head CT scans, Lancet (2018) 2388–2396.

[28] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations - ICLR, 2015.

[29] M. R. Arbabshirani, B. K. Fornwalt, G. J. Mongelluzzo, J. D. Suever, B. D. Geise, A. A. Patel, G. J. Moore, Advanced machine learning in action: identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration, npj Digital Medicine 1 (1) (2018) 1–7.

[30] D. Sato, S. Hanaoka, Y. Nomura, T. Takenaga, S. Miki, T. Yoshikawa, N. Hayashi, O. Abe, A primitive study on unsupervised anomaly detection with an autoencoder in emergency head CT volumes, in: Medical Imaging 2018: Computer-Aided Diagnosis, 2018, p. 60.

[31] M. Monteiro, V. F. J. Newcombe, F. Mathieu, K. Adatia, K. Kamnitsas, E. Ferrante, T. Das, D. Whitehouse, D. Rueckert, D. K. Menon, B. Glocker, Multiclass semantic segmentation and quantification of traumatic brain injury lesions on head CT using deep learning: an algorithm development and multicentre validation study, The Lancet Digital Health 2 (6) (2020) e314–e322.

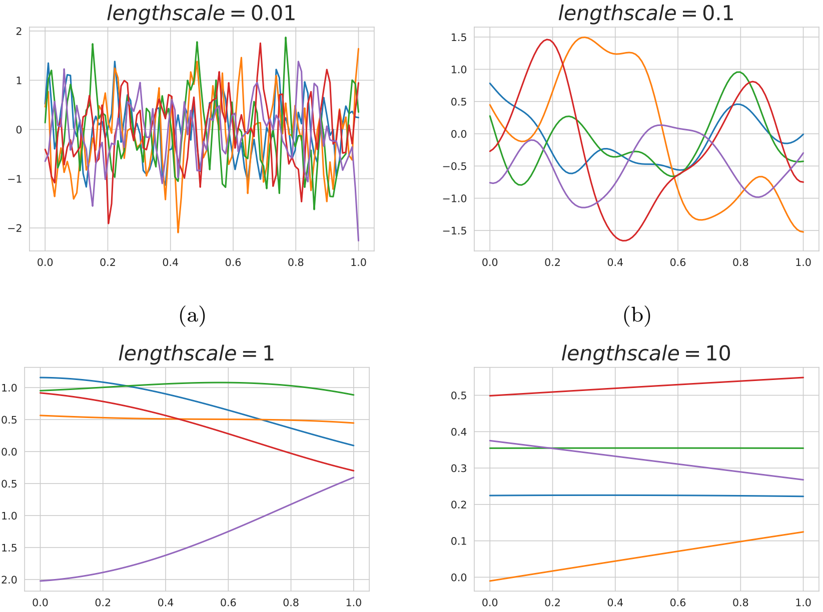
[32] N. T. Nguyen, D. Q. Tran, N. T. Nguyen, H. Q. Nguyen, A CNN-LSTM architecture for detection of intracranial hemorrhage on CT scans, Medical Imaging with Deep Learning (MIDL) (2020).

[33] Y. Zhu, L. Tong, S. R. Deshpande, M. D. Wang, Improved prediction on heart transplant rejection using convolutional autoencoder and multiple instance learning on whole-slide imaging, in: International Conference on Biomedical Health Informatics (BHI), 2019, pp. 1–4.

[34] M. K. Titsias, Variational learning of inducing variables in sparse gaussian processes., in: Artificial Intellgince and Statistics (AISTATS), Vol. 5, 2009, pp. 567–574.

Appendix A. Revisiting Gaussian Processes

This appendix provides a brief introduction to GPs for binary classification. Let us assume a dataset $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$ composed of N instances with $y_n \in \{0, 1\}$.



(a) (b)

(c) (d)

Figure A.9: Example of different sampled functions from a 1-dimensional GP with an SE kernel. Y-axis represents the value of the sampled function and X-axis the input feature of the GP. We use different values of the lengthscale hyperparameter to show how it affects the resulting functions. Shorter values of the lengthscale l produce wriggly curves while larger values produce flat functions.

A Gaussian process prior assumes a multivariate normal distribution in the latent variable $\mathbf{f} = (f_1, ..., f_N)^\intercal$ given $\mathbf{X}$. This prior distribution is defined by a mean function $\mu(\mathbf{x})$ and a kernel (covariance function) $k(\mathbf{x}, \mathbf{x}')$. The mean function is usually set to $\mathbf{0}$, without losing generality. The kernel encodes the prior belief about the data. In this paper we use the Squared Exponential (SE) kernel. It is a common choice in Gaussian Processes due to its flexibility and expressiveness. Also, it encodes smoothness in the latent function, which is a desirable property in many different scenarios. The SE kernel is defined as $k_{\text{SE}}(\mathbf{x}_i, \mathbf{x}_j) = C \exp\left(\frac{-||\mathbf{x}_i - \mathbf{x}_j||^2}{2l^2}\right)$, where the parameters C and l are estimated through the learning task. Figure A.9 shows samples of a GP prior with a SE kernel with different values of l. We can see that the level of smoothness relies on the value of l. Large values of l produce flat functions while small values produce less smooth functions. It is worthy noticing that these functions do not have varying levels of smoothness across the data points. This is one of the motivation to use DGPs, e.g., functions with flat areas and abrupt jumps.

Once we have modelled the latent function $\mathbf{f}$ using a GP prior, we have to define the observation model. Our likelihood for binary classification is the Bernoulli distribution, i.e., $\mathrm{p}(y_i|f_i) = \mathrm{Ber}(y_i; \sigma(f_i))$. Here, σ is the sigmoid and $f_i = f(\mathbf{x}_i)$ refers to the value of the latent variable f at the point $\mathbf{x}_i$. The joint density of $\mathbf{y}$ and $\mathbf{f}$ becomes,

$$\mathrm{p}(\mathbf{y}, \mathbf{f}) = \underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n)}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{f})}_{\text{GP prior}}, \tag{A.1}$$

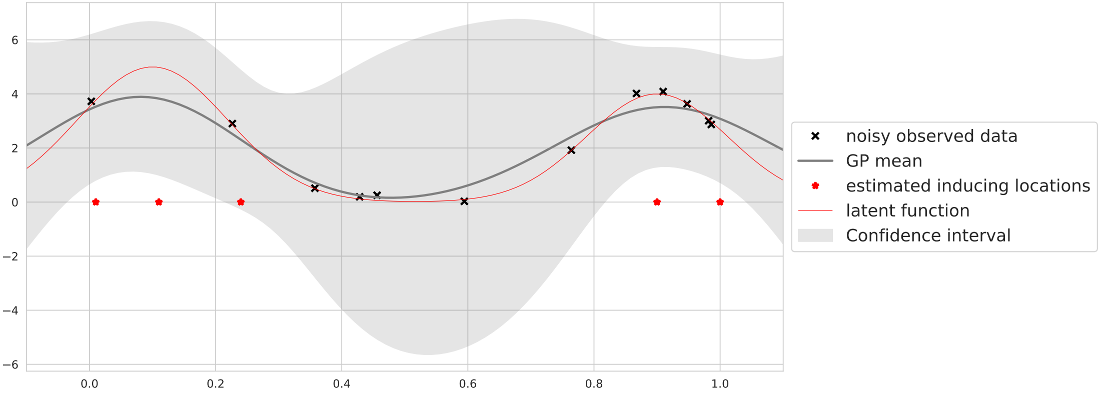


Figure A.10: Example of a Sparse Gaussian Process on a 1-dimensional regression problem. We draw the latent function that generates the noisy observed data, the mean of the estimated GP, the uncertainty and also the estimated inducing locations. The GP has more uncertainty where there are less inducing points.

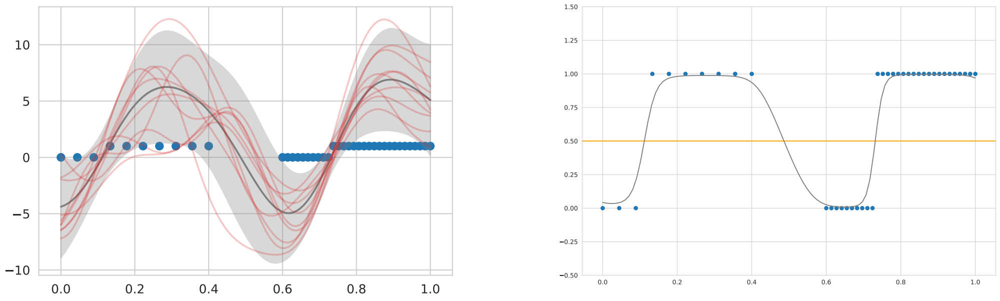
where we assume independence across the instance labels given the latent variables. The goal becomes the estimation of the model parameters, in this case C and l, and the calculation of $\mathrm{p}(\mathbf{f}|\mathbf{y})$.

One main drawback of Gaussian Processes is their scalability. They have computational cost $\mathcal{O}(N^3)$ because their use involves the inversion of an $N \times N$ matrix. To overcome this limitation, sparse GPs have been proposed [34]. The idea behind them is to define $\tilde{M} \ll N$ inducing points u_m which are GP realizations at inducing locations $\mathbf{z}_m$. We can see this as $f(\mathbf{z}) = u$. The inducing points encode the information of the observations in a few points. Their locations $\{\mathbf{z}_m\}_{m=1}^{M}$ are estimated while learning. This approach lightens the computational cost to $\mathcal{O}(n\tilde{M}^2)$. However, the posterior distribution is intractable and approximate inference must be used. The Scalable Variational Gaussian Process (SVGP) inference is the state of the art for sparse GPs [18]. Furthermore, it allows to train in mini-batches. The joint density in this case is given by

$$\mathrm{p}(\mathbf{y}, \mathbf{f}, \mathbf{u}) = \underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n)}_{\text{likelihood}} \underbrace{\mathrm{p}(\mathbf{f}|\mathbf{u}; \mathbf{Z})\mathrm{p}(\mathbf{u}; \mathbf{z})}_{\text{sparse GP prior}}, \tag{A.2}$$

the semicolon notation indicates which are the inputs of each function. The goal here is to calculate $\mathrm{p}(\mathbf{u}, \mathbf{f}|\mathbf{y})$ and estimate the model parameters.

Figure A.10 shows a Sparse Gaussian Process for a 1-dimensional regression problem. We see that the GP mean approaches the latent function that generates the noisy observed data. The latent function is inside the confidence interval, and the uncertainty is larger in areas with less inducing points. Also notice, that the optimal location for the inducing points is where the function has more variations. Figure A.11 shows a GP for binary classification in a 1-dimensional toy problem. In (a), we draw samples for the posterior distribution of $\mathrm{p}(\mathbf{f}|\mathbf{y})$. We can notice that all the samples have the same level of smoothness. Then, in (b) we show the probabilities estimated for the positive class after the sigmoid function.



(a)

(b)

Figure A.11: 1-dimensional binary classification problem with the input dimension on the x-axis and output dimension on the y-axis. The blue points represent the noisy observed data. In (a) we draw the distribution of the latent function p(f_*): the gray line is the mean and the gray shadow the 0.95 confidence interval on the predictions. The classifier has more uncertainty in the region where there are no observations. In (b) we squash the latent function to the [0,1] interval, the black line is p($y_* = 1$).

Appendix A.1. Revisiting Deep Gaussian Processes

A DGP is a hierarchical model which consists of several stacked SVGPs, i.e., the output of a SVGP is the input for the next SVGP [21]. We define $\{\mathbf{F}^l\}_{l=1}^L$ latent variables where each $\mathbf{F}^l$ follows a GP prior with input locations given by $\mathbf{F}^{l-1}$. We consider $\mathbf{F}^0 = \mathbf{X}$. We denote $f_{n,d}^l$ as the latent variable value for the n-th instance in the dimension d (being $1 \leq d \leq D^l$) for the layer l. Notice that in this problem $D^L = 1$. The vector f_n^l contains all the dimensions for the n-th instance in the l-th later. The likelihood of the unobserved instance labels is defined by a Bernoulli distribution,

$$\mathrm{p}(y_n|f_n^L) = \sigma(f_n^L)^{y_n}\left(1 - \sigma(f_n^L)\right)^{1-y_n}, \tag{A.3}$$

Assuming independence across the instance labels given the latent variables, we obtain,

$$\mathrm{p}(\mathbf{Y}|\mathbf{f}^L) = \prod_{n=1}^{N} \mathrm{p}(y_n|f_n^L). \tag{A.4}$$

Because of the computational cost, we have to introduce again the so called sparsity. We have M^{l-1} inducing locations $\mathbf{Z}^{l-1}$ at each layer l with inducing values $\mathbf{U}^l$ for each dimension. So we can write the joint density function,

$$\begin{aligned}\mathrm{p}(\mathbf{Y}, \{\mathbf{F}^l, \mathbf{U}^l\}_{l=1}^L) = &\underbrace{\prod_{n=1}^{N} \mathrm{p}(y_n|f_n^L)}_{\text{likelihood}} \\ &\times \underbrace{\prod_{l=1}^{L} \mathrm{p}(\mathbf{F}^l|\mathbf{U}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1})\mathrm{p}(\mathbf{U}^l; \mathbf{Z}^{l-1})}_{\text{DGP prior}}.\end{aligned} \tag{A.5}$$

The Doubly Stochastic Variational Inference is the state of the art for DGPs [20]. Furthermore, it allows to perform approximate inference and to train in mini-batches.

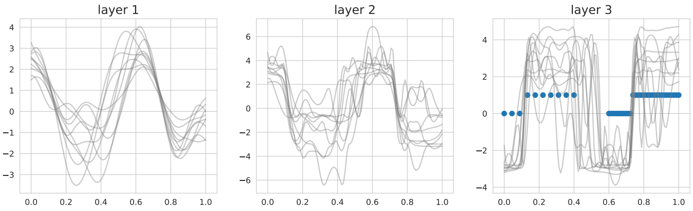


Figure A.12: Samples at every layer of a three-layer DGP trained on a binary toy example. The first two layers are latent spaces where the features are projected onto. The third layer is the output for the final classification. The y-axis represents the values of the latent function before it goes through the sigmoid function. Positive values will be classified as in the positive class and negative values as in the negative one.

Figure A.12 shows samples of the DGP latent function. We show samples from the first and second layer, which are the middle latent representation features before the final classification is done. Then, the third (final) layer is the one that makes the final classification. We can see that the first layer produces smooth functions similar to the ones of the shallow GP. When we apply a GP to these features we can obtain more complex patterns as shown in the second layer. The flat regions are smooth while the jumps in the decision boundaries are abrupter. Although it is a very simple problem, we actually can see the greater expressiveness of DGPs against shallow GPs. This fact encourages their use for complex tasks, as it is in the ICH detection problem.

Appendix B. Detailed DGPMIL inference

This appendix contains all the details for inference in DGPMIL. We follow the doubly stochastic inference to estimate the variational parameters corresponding to the DGP [20]. Together with $\mathbf{Y}_b = \{y_i | i \in \text{bag } b\}$, as defined in section 2.1, we introduce $\mathbf{Y}_{b-n} = \{y_i | y_i \in \text{bag } b \text{ and } i \neq n\}$.

Appendix B.1. Update of $\mathrm{q}(\mathbf{y})$

The optimal $\mathrm{q}(y_n)$ distribution fixing the other distributions is given by

$$\begin{aligned}\log \mathrm{q}(y_n) &= \mathbb{E}_{\mathrm{q}(\mathbf{Y}_{b-n})}\left[\log \mathrm{p}(T_b|\mathbf{y}_b)\right] + \mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log \mathrm{p}(y_n|f_n^L)\right] + \text{const} \\ &= \log H \cdot \mathbb{E}_{\mathrm{q}(\mathbf{Y}_{b-n})}\left[G_b\right] + \mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log \mathrm{p}(y_n|f_n^L)\right] + \text{const}. \end{aligned} \tag{B.1}$$

Now we rewrite the max function as

$$\max \mathbf{Y}_b = y_n + \max \mathbf{Y}_{b-n} - y_n \max \mathbf{Y}_{b-n}, \tag{B.2}$$

and substituting in eq. (B.1) (using also the Jakkola bound [26]) arises

$$\begin{aligned}\log \mathrm{q}(y_n) = {} & y_n \mathbb{E}_{\mathrm{q}(f_n^L)}\left[f_n^L\right] \\ & + y_n \log H (2T_b - 2T_b \mathbb{E}_{\mathrm{q}(\mathbf{Y}_{b-n})}[\max\{\mathbf{Y}_{b-n}\}] \\ & + \mathbb{E}_{\mathrm{q}(\mathbf{Y}_{b-n})}[\max\{\mathbf{Y}_{b-n}\}] - 1) + \text{const}. \end{aligned} \tag{B.3}$$

We use the following approximation as in [16],

$$\mathbb{E}[\max\{y_i\}] \approx \max\{\mathbb{E}[y_i]\}, \tag{B.4}$$

to finally obtain the optimal update for $\mathrm{q}(\mathbf{y})$,

$$\mathrm{q}_n \leftarrow \sigma\left(\mathbb{E}_{\mathrm{q}(f_n^L)}\left[f_n^L\right] + \log H \cdot (2T_b + \max \mathbf{q}_{b-n} - 2T_b \max \mathbf{q}_{b-n} - 1)\right). \tag{B.5}$$

Appendix B.2. ELBO derivation

Using eq. (B.4), the ELBO$(\mathbf{V}, \Theta, \{\mathbf{Z}^{l-1}\}_{l=1}^L)$ is finally approximated by

$$\begin{aligned}
\text{ELBO} &= \sum_{n=1}^N \mathbb{E}_{\mathrm{q}(y_n)\mathrm{q}(f_n^L)}\left[\log \mathrm{p}(y_n|\mathbf{f}_n^L)\right] + \sum_{b=1}^B \sum_{n\in b} \mathbb{E}_{\mathrm{q}(y_n)}\left[\log \frac{H^{G_b}}{H+1}\right] \\
&- \sum_{n=1}^N \mathbb{E}_{q(y_n)}\left[\log \mathrm{q}(y_n)\right] - \sum_{l=1}^L \mathbb{E}_{\mathrm{q}(\mathbf{U}^l)}\left[\log \frac{\mathrm{q}(\mathbf{U}^l)}{\mathrm{p}(\mathbf{U}^l)}\right] \\
&\approx \sum_{n=1}^N q_n \mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log \mathrm{p}(y_n = 1|f_n^L)\right] + (1-q_n)\mathbb{E}_{\mathrm{q}(f_n^L)}\left[\log \mathrm{p}(y_n = 0|f_n^L)\right] \\
&+ \log H \sum_{b=1}^B (2T_b \max \mathbf{q}_b - \max \mathbf{q}_b) \\
&- \sum_{n=1}^N q_n \log q_n + (1-q_n)\log(1-q_n) - \sum_{l=1}^L \mathrm{KL}\left(\mathrm{q}(\mathbf{U}^l)||\mathrm{p}(\mathbf{U}^l)\right) \\
&+ \text{const.}
\end{aligned} \tag{B.6}$$

Appendix B.3. Deep Gaussian Process estimation

We can compute analytically the posterior for $\{\mathbf{F}^l\}_{l=1}^L$ by marginalizing the inducing variables from each layer:

$$\mathrm{q}(\{\mathbf{F}^l\}_{l=1)}^L) = \prod_{l=1}^L \mathrm{q}(\mathbf{F}^l|\mathbf{m}^l, \mathbf{S}^l; \mathbf{F}^{l-1}, \mathbf{Z}^{l-1}) = \prod_{l=1}^L \mathcal{N}(\mathbf{F}^l|\tilde{\boldsymbol{\mu}}^l, \tilde{\boldsymbol{\Sigma}}^l), \tag{B.7}$$

where $[\tilde{\boldsymbol{\mu}}^l]_n = \mu_{\mathbf{m}^l,\mathbf{Z}^{l-1}}(\mathbf{f}_n^{l-1})$ and $[\tilde{\boldsymbol{\Sigma}}^l]_{ij} = \Sigma_{\mathbf{S}^l,\mathbf{Z}^{l-1}}(\mathbf{f}_i^{l-1}, \mathbf{f}_j^{l-1})$. The explicit expression for the mean vector $\tilde{\boldsymbol{\mu}}^l$ and the covariance matrix $\tilde{\boldsymbol{\Sigma}}^l$ can be found in [20, Eqs. (7-8)]. We are able to compute the i-th marginal at each layer because it only depends on the corresponding i-th input of the previous layer. This allows to sample from the last layer $\mathbf{f}_i^L$ by recursively sampling from all the previous layers $\hat{\mathbf{f}}_i^1 \to \hat{\mathbf{f}}_i^2 \to \cdots \to \hat{\mathbf{f}}_i^L$. This can be easily performed by means of univariate Gaussians. We first sample a $\varepsilon_i^l \sim \mathcal{N}(0,1)$ and then for $l = 1, \ldots, L$:

$$\hat{\mathbf{f}}_i^l = \mu_{\mathbf{m}^l,\mathbf{z}^{l-1}}(\hat{\mathbf{f}}_i^{l-1}) + \varepsilon_i^l \cdot \sqrt{\Sigma_{\mathbf{S}^l,\mathbf{z}^{l-1}}(\hat{\mathbf{f}}_i^{l-1}, \hat{\mathbf{f}}_i^{l-1})}. \tag{B.8}$$

Since we can sample from the posterior distribution in the last layer, the expectation $\mathbb{E}_{\mathrm{q}(f_n^L)}[\log \mathrm{p}(y_n|f_n^L)]$ in the ELBO (see eq. (B.6)) can be approximated with a Monte Carlo sample generated with eq. (B.8). Similarly, we can compute the expectation $\mathbb{E}_{\mathrm{q}(f_n^L)}[f_n^L]$ in the update of the $\mathrm{q}(\mathbf{Y})$, see eq. (B.5). For scalability, we can use mini-batches in the optimization since the ELBO factorizes across data points.

Once the model is trained and the ELBO optimized, we can make a prediction for new test point $\mathbf{X}_*$. For this, we sample S times from the posterior using eq. (B.8). In this case, we use the test location as initial input. This yields a set $\{\mathbf{f}_*^{L-1}(s)\}_{s=1}^S$ with S samples. Then, the density over f_*^L is given by the Gaussian mixture (recall that all the terms in eq. (B.7) are Gaussian):

$$\mathrm{q}(f_*^L) = \frac{1}{S}\sum_{s=1}^S \mathrm{q}(f_*^L|\mathbf{m}^L, \mathbf{S}^L; \mathbf{f}_*^{L-1}(s), \mathbf{Z}^{L-1}). \tag{B.9}$$

Appendix C. Additional results

Here, we report additional tables with results. These tables are commented in the main text but we included them here for better readability.

Table C.1: Mean results for 5 different runs of DGPMIL2 with 8-dimensional input features. The results are for both RSNA and CQ500 datasets. We study the metrics for a varying number of inducing points $\tilde{M}$.

	$\tilde{M}$	F1 score	AUC-ROC	AUC-PR
RSNA	10	0.829±0.018	0.953±0.012	0.954±0.014
	50	0.834±0.016	0.954±0.01	0.96±0.008
	200	0.839±0.006	0.957±0.011	0.961±0.011
	500	0.835±0.006	0.956±0.012	0.962±0.009
CQ 500	10	0.714±0.02	0.899±0.01	0.853±0.026
	50	0.734±0.024	0.911±0.012	0.887±0.024
	200	0.735±0.022	0.909±0.005	0.889±0.011
	500	0.731±0.026	0.913±0.01	0.893±0.009

Table C.2: Mean results for 5 different runs of DGPMIL2 with 8-dimensional input features. The results are for both RSNA and CQ500 datasets. We study the metrics for a varying number of dimensions D in the latent space.

	D	F1 score	AUC-ROC	AUC-PR
RSNA	3	0.839±0.006	0.957±0.011	0.961±0.011
	10	0.837±0.008	0.957±0.09	0.964±0.006
	50	0	0.5±0	0.48±0
CQ 500	3	0.735±0.022	0.909±0.005	0.889±0.011
	10	0.733±0.022	0.914±0.013	0.902±0.0279
	50	0	0.5±0	0.418±0

Chapter 6

Conclusions and future work

6.1 Conclusions

In this thesis, we have shown that GPs and DGPs can outperform DL methods for different labeling paradigms (i.e., supervised and weakly supervised learning) and domains (i.e., volcanology and medicine). In these databases, the suitability of GP methods is remarkable. The main findings of this thesis are as follows:

- Regarding seismic classification, GPs, and DGPs outperformed DL methods performing much better at detecting rare classes. Also, GPs and DGPs estimated uncertainty better giving more accurate probabilities to the predictions.

- Regarding prostate cancer detection, we showed that features extracted from the Optical Density space encoded more relevant information. Also, morphological and texture features achieved state-of-the-art results when classifying them with a GP or DGP. We showed that GPs and DGPs outperformed every other shallow classifier, and also, they were competitive with DL methods. Finally, we empirically proved that GPs and DGPs are more efficient than DL methods.

- Regarding crowdsourcing classification in cancer, a GP trained with deep features extracted from a pretrained deep neural network performed better than DL methods. GPs for crowdsourcing automatically modeled the noisy labels and the expertise of each annotator. This model, trained with noisy labels, was competitive with the one trained with expert annotations in breast cancer classification. We observed that crowdsourcing is a feasible solution to the lack of labeled data, since massively cancer images can be annotated engaging medical students.

- Regarding MIL in ICH detection, the proposed DGPMIL achieved better results than DL and shallow GPs. We showed the need for hierarchical models based on GPs to learn complex functions in real applications. This model performed better both at scan and slice levels, and its precision was remarkably better. It was able to identify better the false positives being a more robust classifier for medicine use.

These results open a new door for efficient labeling and the possibility of training more powerful models.

6.2 Future work

This thesis also opens new interesting research problems to be addressed in the future. We list the main ones here:

- Crowdsourcing in volcanology. The lack of labeled databases can be tackled with crowdsourced labels. We will explore how to replicate this process when seismic signals are reported from different stations and annotated by several participants with varying expertise.
- Improving the performance of GP for histopathology classification. Since GPs can not deal with feature extraction, training the CNN and the GP in an end-to-end manner might lead to better results.
- Improving the efficiency in labeling. A new problem to be addressed is the scenario where non-pathologists provide coarse-grained labels. It will combine both crowdsourcing and multiple instance learning.
- Improving the interpretability of predictions in crowdsourcing. Pixel-wise predictions are more insightful than global ones. We will study how to segment histopathological images from crowdsourcing tasks.

References

Ahmed, S., Shaikh, A., Alshahrani, H., Alghamdi, A., Alrizq, M., Baber, J., & Bakhtyar, M. (2021). Transfer learning approach for classification of histopathology whole slide images. *Sensors*, *21*(16), 5361.

Bishnoi, S., Ravinder, R., Grover, H. S., Kodamana, H., & Krishnan, N. A. (2021). Scalable gaussian processes for predicting the optical, physical, thermal, and mechanical properties of inorganic glasses with large datasets. *Materials Advances*, *2*(1), 477–487.

Campanella, G., Hanna, M. G., Geneslaw, L., Miraflor, A., Werneck Krauss Silva, V., Busam, K. J., ... Fuchs, T. J. (2019). Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, *25*(8), 1301–1309.

Cho, J., Park, K.-S., Karki, M., Lee, E., Ko, S., Kim, J. K., ... Park, S. (2019). Improving sensitivity on identification and delineation of intracranial hemorrhage lesion using cascaded deep learning models. *Journal of Digital Imaging*, *32*(3), 450–461.

Damianou, A., & Lawrence, N. (2013, 29 Apr–01 May). Deep Gaussian processes. In *Proceedings of the sixteenth international conference on artificial intelligence and statistics* (Vol. 31, pp. 207–215). Scottsdale, Arizona, USA: PMLR.

Ferlaino, M., Glastonbury, C. A., Motta-Mejia, C., Vatish, M., Granne, I., Kennedy, S., ... Nellåker, C. (2018). Towards deep cellular phenotyping in placental histology. *CoRR*, *abs/1804.03270*.

Haußmann, M., Hamprecht, F. A., & Kandemir, M. (2017). Variational bayesian multiple instance learning with gaussian processes. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6570–6579).

Hensman, J., de G. Matthews, A. G., & Ghahramani, Z. (2015). Scalable variational gaussian process classification. In *Artificial intellgince and statistics (AISTATS)* (Vol. 38).

Jnawali, K., Arbabshirani, M. R., Rao, N., & M.d, A. A. P. (2018). Deep 3d convolution neural network for CT brain hemorrhage classification. In *Medical imaging 2018: Computer-aided diagnosis* (Vol. 10575, p. 105751C). International Society for Optics and Photonics.

Kandemir, M. (2015, 07–09 Jul). Asymmetric transfer learning with deep gaussian processes. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd international conference on machine learning* (Vol. 37, pp. 730–738). Lille, France: PMLR.

Karimi, D., Dou, H., Warfield, S. K., & Gholipour, A. (2020). Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical Image Analysis*, *65*, 101759.

Kim, H., Yoon, H., Thakur, N., Hwang, G., Lee, E. J., Kim, C., & Chong, Y. (2021). Deep learning-based histopathological segmentation for whole slide images of colorectal cancer in a compressed domain. *Scientific reports*, *11*(1).

Koné, I., & Boulmane, L. (2018). Hierarchical resnext models for breast cancer histology image classification. In *Image analysis and recognition* (pp. 796–803). Cham: Springer International Publishing.

Li, Y., Rao, S., Hassaine, A., Ramakrishnan, R., Canoy, D., Salimi-Khorshidi, G., ... Rahimi, K. (2021). Deep bayesian gaussian processes for uncertainty estimation in electronic health records. *Scientific reports*, *11*(1), 1–13.

Morales-Álvarez, P., Ruiz, P., Coughlin, S., Molina, R., & Katsaggelos, A. (2022, March). Scalable variational gaussian processes for crowdsourcing: Glitch detection in ligo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(3), 1534-1551.

Murphy, K. P. (2022). *Probabilistic machine learning: An introduction.* MIT Press.

Nir, G., Hor, S., Karimi, D., Fazli, L., Skinnider, B. F., Tavassoli, P., ... Salcudean, S. E. (2018). Automatic grading of prostate cancer in digitized histopathology images: Learning from multiple experts. *Medical Image Analysis*, *50*, 167 - 180.

Phong, T. D., Duong, H. N., Nguyen, H. T., Trong, N. T., Nguyen, V. H., Van Hoa, T., & Snasel, V. (2017). Brain hemorrhage diagnosis by using deep learning. In *International conference on machine learning and soft computing* (pp. 34–39).

Priego-Torres, B. M., Sanchez-Morillo, D., Fernandez-Granero, M. A., & Garcia-Rojo, M. (2020). Automatic segmentation of whole-slide h&e stained breast histopathology images using a deep convolutional neural network architecture. *Expert Systems With Applications*, *151*, 113387.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning (adaptive computation and machine learning).* The MIT Press.

Salimbeni, H., & Deisenroth, M. (2017). Doubly stochastic variational inference for deep gaussian processes. In *Advances in neural information processing systems 30* (pp. 4588–4599). Curran Associates, Inc.

Shallu, & Mehra, R. (2018). Breast cancer histology images classification: Training from scratch or transfer learning? *ICT Express*, *4*(4), 247 - 254.

Strub, W. M., Leach, J. L., Tomsick, T., & Vagal, A. (2007). Overnight preliminary head CT interpretations provided by residents: Locations of misidentified intracranial hemorrhage. *American Journal of Neuroradiology*, *28*(9), 1679–1682.

Svendsen, D. H., Martino, L., & Camps-Valls, G. (2020). Active emulation of computer codes with gaussian processes – application to remote sensing. *Pattern Recognition*, *100*, 107103.

Svendsen, D. H., Morales-Álvarez, P., Ruescas, A. B., Molina, R., & Camps-Valls, G. (2020). Deep gaussian processes for biogeophysical parameter retrieval and model inversion. *ISPRS Journal of Photogrammetry and Remote Sensing*, *166*, 68-81.

Titano, J. J., Badgeley, M., Schefflein, J., Pain, M., Su, A., Cai, M., ... Oermann, E. K. (2018). Automated deep-neural-network surveillance of cranial images for acute neurologic events. *Nature Medicine*, *24*(9), 1337–1341.

Titos, M., Bueno, A., Garcia, L., & Benitez, C. (2018). A deep neural networks approach to automatic recognition systems for volcano-seismic events. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *11*(5), 1533–1544.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse gaussian processes. In *Artificial intellgince and statistics (AISTATS)* (Vol. 5, p. 567-574).

Wu, Y., Schmidt, A., Hernández-Sánchez, E., Molina, R., & Katsaggelos, A. K. (2021). Combining attention-based multiple instance learning and gaussian processes for CT hemorrhage detection. In *Medical image computing and computer assisted intervention – MICCAI* (pp. 582–591).

Yang, H., Chen, L., Cheng, Z., Yang, M., Wang, J., Lin, C., ... others (2021). Deep learning-based six-type classifier for lung cancer and mimics from histopathological whole slide images: a retrospective study. *BMC medicine*, *19*(1), 1–14.

Zhou, Z.-H. (2018). A brief introduction to weakly supervised learning. *National Science Review*, *5*, 44-53.