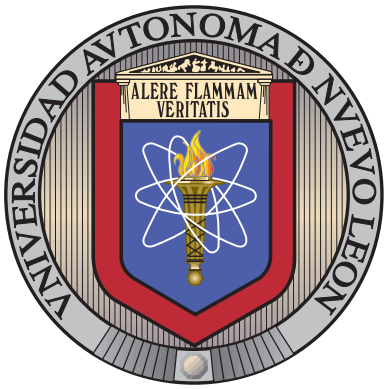
Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado



Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos

por

Mario Alberto Saucedo Espinosa

en opción al grado de

Maestro en Ciencias

en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León mayo 2012

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos

por

Mario Alberto Saucedo Espinosa

en opción al grado de

Maestro en Ciencias

en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León mayo 2012

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis «Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos», realizada por el alumno Mario Alberto Saucedo Espinosa, con número de matrícula 1272375, sea aceptada para su defensa como opción al grado de Maestro en Ciencias en Ingeniería de Sistemas.

El Comité de Tesis

Dr. J. Arturo Berrones Santos
Asesor

Dr. Óscar L. Chacón Mondragón
Revisor

Dr. Víctor M. Treviño Alvarado
Revisor

Vo. Bo.

Dr. Moisés Hinojosa Rivera
División de Estudios de Posgrado

San Nicolás de los Garza, Nuevo León, mayo 2012

A mi madre, Adrianita, que me levantó de mis valles más obscuros sin perder por un sólo día la fe en mí. Hoy no estaría aquí si no fuera por ti.

A mi padre, Mario, que en ningún momento ha dejado de ser mi modelo a seguir, heredándome su tenacidad, su inteligencia y su honestidad. Mi amor por la ingeniería te lo debo a ti.

A mi hermano, José Guillermo, fiel compañero de tantas noches de desvelo y pláticas motivadoras. Siempre has sido mi mejor amigo.

A mi princesa, Brendita, que me enseñó que existe un mundo maravilloso más allá del estudio, a su lado. Me diste tu apoyo y tu ternura cuando más lo necesitaba.

Índice general

Índice de figuras

Índice de tablas

Agradecimientos

Agradezco profundamente a mis padres, Mario y Adrianita, y a mis hermanos, Pepe y Catalina, por siempre creer en mí. Por su comprensión y apoyo en los momentos malos y menos malos que he vivido durante toda mi formación profesional. En gran parte es gracias a ustedes que hoy puedo ver cumplida esta meta. Agradezco también a mi princesa, Brenda Ayala, por su paciencia, su comprensión y su amor durante esta aventura como estudiante de posgrado. A mis grandes amigos: Juan, Érik, Manuel, Verástegui, Francisco, Jacob y Ernesto, que durante todo mi trayecto universitario nunca han dejado de alentarme.

Agradezco de manera muy especial al Dr. J. Arturo Berrones Santos por aceptar el reto de fungir como mi director de tesis, por compartir conmigo sus invaluables conocimientos y su orientación por medio de un sinfín de pláticas interesantes, por integrarme a muchos de sus proyectos de investigación y por darme la confianza para trabajar de una manera sostenible a lo largo de este proyecto en particular, siempre con un trato humano y amable hacia sus estudiantes. Gracias por haber sido partícipe de mi formación como investigador. De igual forma, agradezco al Dr. Óscar L. Chacón Mondragón por haberse tomado el tiempo para participar en mi formación académica como miembro de mi Comité de Tesis, por tener siempre abierta su oficina cuando tuve alguna duda, tanto profesional como personal, por siempre tener un consejo o solución para cada planteamiento, por sus ánimos y palabras alentadoras. Un agradacimiento especial al Dr. Víctor M. Treviño Alvarado por formar parte también de mi Comité de Tesis y tomarse el tiempo para revisar y mejorar este manuscrito.

Agradezco también de manera muy especial a la Dra. Yasmín A. Ríos Solís, por brindarme la oportunidad de formar parte de su equipo de trabajo en diversos proyectos durante gran parte de mis estudios de maestría. Por siempre impulsarme a superar la calidad de mi investigación, de mis proyectos y mis trabajos, por su trato sincero y humano, por su invaluable apoyo durante mi primera experiencia como expositor en un congreso internacional y por siempre animarme a continuar mis estudios. Hago extensivo este agradecimiento a toda la comunidad PISIS, especialmente a sus profesores, por siempre estar abiertos a diálogos de los más variados e interesantes temas. Igualmente, a mis nuevos amigos producto de mi tiempo en PISIS.

Un agradecimiento al Dr. Franco Bagnoli de la Università degli Studi di Firenze y a todo su equipo de trabajo, por el apoyo, el interés y las facilidades proporcionadas para la realización de una estancia académica de investigación en Italia, la cual fue una parte importante de la investigación aquí presentada.

Mi más profundo agradecimiento al Consejo Nacional de Ciencia y Tecnología por la beca de manutención económica brindada durante la realización de este proyecto, además del apoyo económico proporcionado dentro del Programa de Becas Mixtas que me permitió realizar la estancia de investigación en Italia. Finalmente, mi eterno agradecimiento a la Universidad Autónoma de Nuevo León, esta institución de enorme calidad académica que me ha formado como profesionista, y a su Facultad de Ingeniería Mecánica y Eléctrica por brindarme el apoyo para mis estudios profesionales al otorgarme una beca de colegiatura durante la totalidad de mis estudios de maestría.

Resumen

Mario Alberto Saucedo Espinosa.

Candidato para el grado de Maestro en Ciencias
con especialidad en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.
Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio:

Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos

Número de páginas: 134.

Objetivos y método de estudio: El objetivo principal de esta tesis consiste en evaluar la capacidad de predicción que tienen las redes neuronales Bayesianas, los procesos Gaussianos y el modelo de mezcla infinita de Gaussianas sobre problemas de estudio que involucran altos niveles de ruido y no-linealidad conforme el tamaño del conjunto de entrenamiento disminuye. Al reducir el número de observaciones disponibles para la etapa de entrenamiento, el efecto del ruido y la no-linealidad tienden a volverse más dominantes [46], lo que afecta considerablemente la capacidad de pronóstico de las técnicas basadas en la estadística frecuentista al aumentar el riesgo de entrenar un modelo con sobreajuste. Para evitar el sobreajuste se necesita

acoplar al entrenamiento alguna metodología que regule la complejidad del modelo. Los métodos Bayesianos contienen de manera implícita un mecanismo que regula tal complejidad, por lo que se espera que sus pronósticos sean más asertivos que aquellos provenientes de métodos frecuentistas, por lo que se realiza una comparación con el *boostrap*, una técnica que ha mostrado resultados precisos en problemas donde otros métodos han fallado. Un objetivo colateral de esta tesis es la implementación computacional de los métodos de solución propuestos.

CONTRIBUCIONES Y CONLUSIONES: En esta tesis se hizo énfasis en el estudio de un método que representa algunas ventajas por encima de los métodos clásicos de aprendizaje estadístico y computacional pero que no ha sido aplicado de manera continua en la literatura: el modelo de mezclas infinitas de Gaussianas. Adicionalmente, el desempeño de éste, de las redes neuronales Bayesianas y de los procesos Gaussianos es comparado con el desempeño del *bootstrap*. El impacto que tiene el tamaño del conjunto de entrenamiento en el desempeño de estos métodos es también estudiado, especialmente la variación en la capacidad de pronóstico al disminuir la cantidad de observaciones para el entrenamiento del modelo.

Se mostró que el modelo de mezcla infinita de Gaussianas es una técnica general robusta en cuanto al tamaño del conjunto de entrenamiento, destacándose de entre los demás métodos de solución por obtener las mejores capacidades de pronóstico en todos los problemas de estudio y con todos los conjuntos de entrenamiento. La diferencia promedio en la capacidad de pronóstico es considerable en el problema del XOR continuo, mientras que esta diferencia promedio con respecto al resto de los métodos parece no serlo en la inferencia de la afinidad de los acoplamientos enzimáticos, aunque para ciertos tamaños muestrales sí representa una marcada ventaja. Una cuestión similar sucede para la inferencia de concentración de metales pesados, donde además la mezcla infinita de Gaussianas muestra una capacidad de pronóstico considerablemente mejor en comparación con algoritmos del estado del arte presentados en [2]. Los métodos Bayesianos contienen de manera implícita un mecanismo que previene el sobreajuste, y esto ha quedado evidenciado dado que

los tres métodos Bayesianos presentan errores cuadrados medios normalizados relativamente pequeños. Sin embargo, el *bootstrap* ha demostrado ser una herramienta adecuada para evitar el sobreajuste, al arrojar resultados comparables a aquellos del modelo de mezcla infinita de Gaussianas, e incluso capacidades de pronóstico que superan las de un método clásico como lo es la red neuronal. Comparando las redes neuronales con el modelo de mezcla infinita de Gaussianas, se encuentra que existe una importante diferencia entre la capacidad de pronóstico de ambos métodos cuando el tamaño del conjunto de entrenamiento es pequeño, aminorándose esta diferencia conforme se aumenta el tamaño muestral. No obstante, se muestra que los cuatro métodos de estudio son capaces de modelar adecuadamente el ruido y la no-linealidad de los conjuntos de observaciones multidimensionales, previniendo el sobreajuste. Además de la robustez empírica que presenta, entre las ventajas de utilizar el modelo de mezcla infinita de Gaussianas se encuentran: (i) el modelo es capaz de realizar clasificación y regresión sin modificar su estructura, (ii) no tiene parámetros que necesiten de un ajuste y (iii) no necesita información externa a priori de sus parámetros para realizar inferencia Bayesiana. Nuestro interés en este método consiste en que es un algoritmo completamente automático que aprende eficazmente la cantidad de componentes y las Gaussianas que modelan sus observaciones, además de asignar un grado de *responsabilidad* que tiene cada Gaussiana para una observación. Esta es una propiedad sumamente interesante para el aprendizaje en línea. Para llevar a cabo estos objetivos se implementaron computacionalmente los tres métodos Bayesianos en el lenguaje *R*, y se seleccionaron problemas de estudio que permitieran un análisis interpretativo de la variación en la capacidad de pronóstico de los métodos. Los tres problemas presentados en este estudio son problemas no-lineales retadores provenientes de diferentes disciplinas, teniendo características diferentes. Por otro lado, entre las contribuciones aportadas en este estudio se encuentran:

- Se comprobó la efectividad de los métodos Bayesianos bajo estudio. Se comprobó además un método no-Bayesiano que, de acuerdo a su popularidad y su capacidad para resolver problemas en donde otros métodos fallan, logró un

desempeño comparable al modelo de mezcla infinita de Gaussianas y en ocasiones superior a las redes neuronales.

- Se demostró el potencial que tienen los métodos Bayesianos, especialmente la mezcla infinita de Gaussianas, para evitar el sobreajuste conforme el tamaño muestral decrece.

- Se aplicó el modelo infinito de mezcla de Gaussianas a problemas retadores de interés actual, con lo que se enriquece el bajo número de aplicaciones de este método que han sido estudiadas.

- Se implementaron las redes neuronales y los procesos Gaussianos con aprendizaje Bayesiano en *R*, al estar sólo disponibles en otros lenguajes.

- Se implementó el modelo de mezcla infinita de Gaussianas, cuya versión multivariada no se encuentra disponible en ningún lenguaje, hasta donde nosotros sabemos. Esta implementación es de importancia para estudios que se derivan de éste y de otros que son ajenos, por ejemplo, en materia de aprendizaje en línea.

- Se escribió un apéndice que detalla la implementación computacional de los métodos Bayesianos, una aportación importante para comprender la forma de operar del modelo de mezcla infinita de Gaussianas y para esquivar las complicaciones que implica su implementación. Esta contribución es importante, al no haber un documento que detalle tal implementación.

Firma del asesor: _____

Dr. J. Arturo Berrones Santos

Capítulo 1

Introducción

Este documento describe la tesis de investigación «Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos», en la cual se busca entrenar modelos no-lineales mediante métodos Bayesianos que sirvan como herramientas de predicción para problemas retadores de interés provenientes de diferentes disciplinas, especialmente en el caso cuando se tiene un conjunto pequeño de observaciones. Se busca además comparar el desempeño de su capacidad de pronóstico contra una técnica que ha mostrado gran aceptación y popularidad en la estadística aplicada: el método *bootstrap*, al proporcionar resultados satisfactorios en problemas donde otros métodos han fallado [65]. Los niveles de ruido y no-linealidad inherentes a los conjuntos de observación estudiados en esta tesis aumentan el riesgo de entrenar un modelo con sobreajuste, especialmente cuando se tienen pocas observaciones para la fase de entrenamiento [55, 71, 22]. Este efecto será estudiado para determinar la forma en que el ruido y la no-linealidad afectan la capacidad de pronóstico de las técnicas en función del tamaño del conjunto de entrenamiento. Una descripción más extensa del problema que abarca esta tesis se presenta en la Sección 1.1, mientras que en la Sección 1.2 se proporciona una discusión acerca de la motivación para realizar este estudio. Por su parte, la Sección 1.3 resume los objetivos formales que persigue esta investigación. Finalmente, en la Sección 1.4 se muestra la forma en que está estructurada el resto de esta tesis.

1.1 Descripción del Problema

Cuando se desea estudiar una determinada característica de una población en concreto, como puede ser una propiedad física, una característica química o la velocidad de propagación de una enfermedad, no es costeable trabajar con toda la población bajo estudio por cuestiones de tiempo y economía. En su lugar se toma una muestra aleatoria que se supone es representativa de la población (i.e., no existen sesgos de selección en la obtención de la muestra), la cual proviene de un proceso afectado por variabilidad aleatoria (i.e., por ruido). La inferencia estadística es el conjunto de técnicas y metodologías estadísticas mediante las cuales se realizan estimaciones sobre una población a partir de una muestra aleatoria observada [14]. Actualmente existen un gran número de técnicas de pronóstico basadas en la estadística clásica, también conocida como estadística frecuentista, en donde los parámetros a estimar de un modelo son considerados valores desconocidos, pero fijos [47]. Los estimadores que se construyen para estos parámetros corresponden con aquellos que maximizan la probabilidad de tener la muestra aleatoria observada. Sin embargo, la aplicación de la estadística frecuentista presenta algunas desventajas. Una de las más importantes es el riesgo de aprender un modelo con sobreajuste, el cual aparece con mayor frecuencia conforme el tamaño del conjunto de entrenamiento decrece [12, 47]. El sobreajuste se presenta cuando un modelo no es capaz de diferenciar el ruido de las observaciones en el conjunto de entrenamiento y éste ajusta sus parámetros de tal forma que aprende tanto las observaciones como el ruido, tratándolo como parte de una no-linealidad inherente al conjunto de observaciones. Si el modelo se entrena nuevamente con un conjunto de observaciones diferente, se espera que los parámetros estimados varíen en una proporción relacionada al ruido, donde nuevamente el modelo asigna el ruido a una no-linealidad en las observaciones [41]. Esta situación conduce a que el modelo tenga un bajo nivel predictivo para observaciones fuera del conjunto de entrenamiento. Para disminuir el riesgo del sobreajuste generalmente se aumenta el tamaño del conjunto de entrenamiento, aunque esto no es siempre una solución costeable, o incluso factible [12].

Existen además una serie de métodos cuya función es disminuir el riesgo de aprender un modelo con sobreajuste [32, 47, 56, 68, 16, 5]. Una de tales metodologías consiste en emplear la inferencia Bayesiana, la cual es una corriente estadística que se diferencia de la frecuentista por considerar los parámetros de un modelo como variables aleatorias [5], de tal forma que es posible asociar distribuciones de probabilidad a estos parámetros para representar la incertidumbre que se tiene acerca de su valor. En contraposición a los métodos frecuentistas, es bien conocido que los métodos Bayesianos contienen de manera implícita un mecanismo que regula la complejidad del modelo y por ende previene el sobreajuste [12]. El avance tecnológico de los últimos años ha ocasionado una revolución en la aplicación de técnicas Bayesianas, las cuales requieren de cálculos computacionalmente pesados que hasta hace algunos años no era costeable desarrollar, de modo que la tendencia actual es utilizar técnicas Bayesianas por encima de las frecuentistas para labores de inferencia [5], no sólo por la prevención que exhibe en cuestiones de sobreajuste, sino porque la aplicación de la teoría de probabilidad permite extraer conclusiones más intuitivas.

Regular la complejidad del modelo es un aspecto esencial en el aprendizaje computacional y estadístico, el cual toma lugar debido a la combinación del ruido y la no-linealidad que forman parte de una gran proporción de los fenómenos físicos. Las reacciones químicas, la mecánica de fluidos, la dinámicas de gases, la elasticidad, la combustión y muchos otros fenómenos están gobernados por funciones no-lineales. Es por esta razón que gran parte del diseño moderno de métodos estadísticos y de aprendizaje automático se dedica al análisis de los sistemas no-lineales. Un sistema no-lineal es todo aquel sistema en que sus características o propiedades no pueden modelarse como una combinación lineal de componentes independientes [30]. No obstante, muchos sistemas físicos tienen bajos niveles de no-linealidad, en el sentido que los términos lineales tienden a dominar el sistema, a pesar que los términos no-lineales juegan un papel importante. Una primera aproximación para estos casos son los modelos lineales [41]. Por el contrario, cuando son los términos no-lineales los que tienden a dominar el sistema, la tarea de entrenar satisfactoriamente un modelo se complica. La posibilidad de múltiplos óptimos locales y la búsqueda en

funciones no-diferenciables son sólo algunos de los problemas que debe enfrentar el proceso de optimización que demanda la estadística frecuentista [8]. Además, como se describió anteriormente, el efecto combinado del ruido y la no-linealidad afecta la capacidad de pronóstico del modelo conforme el tamaño del conjunto de entrenamiento disminuye [12]. En esta tesis se aplican métodos Bayesianos estadísticos y de aprendizaje automático no-lineales para analizar el efecto que tienen el ruido y la no-linealidad en el conjunto de observaciones conforme el tamaño del conjunto de entrenamiento disminuye. Los problemas de estudio considerados tienen diferentes niveles de dificultad, que abarcan un alto nivel de no-linealidad y correlación en los atributos de salida, por mencionar algunos, además de provenir de ambientes en donde el ruido forma parte de las observaciones. Un interés especial que se persigue en esta tesis es evaluar la capacidad de predicción de los diferentes métodos Bayesianos cuando el conjunto de entrenamiento es pequeño, y comparar tales resultados con un método que ha sido aplicado recientemente en problemas donde otros métodos fallan, mostrando resultados precisos: el método *bootstrap*. Para un tamaño muestral pequeño, los niveles de ruido y no-linealidad presentes en los diferentes conjuntos de observaciones aumentan el riesgo de entrenar un modelo con sobreajuste, por lo que esta comparación resulta interesante. Para realizar estas pruebas y comparaciones computacionales se implementaron los métodos Bayesianos de solución descritos en esta tesis.

1.2 Motivación y Justificación

Los problemas de estudio propuestos son problemas retadores provenientes de diferentes áreas del conocimiento, como lo son la electrónica [55], la bioquímica y biología molecular [71] y las geociencias [22]. Dos de estos casos corresponden con problemas de interés actual. Los problemas de estudio tienen una naturaleza no-lineal, además de provenir de ambientes en donde las observaciones están sujetas a ruido. Es común que en las aplicaciones de regresión existan diversas fuentes de ruido en las observaciones [53], las cuales pueden estar relacionadas con el muestreo,

como resultado de representar la población mediante una muestra aleatoria [35], o ser ajenas a éste, de entre las cuales sobresale el ruido de medición. El ruido muestral consiste en la selección de una muestra que es poco (o nada) representativa de la población, y la derivación de conclusiones basadas en ésta. Por otro lado, el error de medición es inherente al proceso de medición y a las limitaciones del instrumento con que se realizan las mediciones. A la combinación de ambas fuentes de ruido se le denomina simplemente como ruido en este contexto. El objetivo de aplicar técnicas de aprendizaje automático es que el algoritmo modele los patrones que realmente ocurren en las observaciones y que reconozca e ignore el ruido en ellas.

Cada problema de estudio ha sido cuidadosamente seleccionado en base a sus características. En el problema XOR continuo (Sección 3.1) se tienen pares ordenados como atributos de entrada, mientras que la salida es una variable booleana proporcionada por una compuerta lógica digital. La salida de la compuerta está basada en reglas lógicas sencillas, pero que al mismo tiempo son complicadas de aprender para un sistema de aprendizaje artificial [55]. Las características más importantes de este problema consisten en que (i) se conoce la función óptima de discriminación para los pares ordenados, y (ii) las observaciones están libres de ruido, de modo que la variación en la capacidad de pronóstico de un modelo conforme se disminuye el tamaño muestral es únicamente por efecto de la no-linealidad presente en las observaciones. En el segundo problema (Sección 3.2) se entrena un modelo para pronosticar la afinidad que presenta una enzima por un sustrato, lo que implica estimar el nivel de interacción en un complejo enzimático [34]. La función que se desea aproximar es altamente no-lineal [44], lo que hace de éste un problema complicado para cualquier método de aprendizaje computacional y estadístico. Aunado a esto, se espera que las observaciones presenten ruido por tratarse de observaciones biológicas. Por ejemplo, los complejos con baja afinidad no pueden ser fácilmente distinguibles del ruido. De esta forma, se esperan extraer conclusiones interesantes acerca del efecto combinado del ruido y la no-linealidad, y de cómo este efecto impacta la capacidad de pronóstico de los métodos de solución como función del tamaño del conjunto de entrenamiento. Finalmente, en el tercer problema de estudio se aborda un problema

de contaminación de la superficie terrestre, en donde se intenta inferir la concentración de diversos metales pesados potencialmente tóxicos depositados en el suelo [22]. Medir de manera directa la concentración de un metal puede ser un proceso costoso, como es el caso del cobre, por lo que se prefiere predecir su concentración mediante mediciones más accesibles, como la concentración de otros metales. Este problema consiste en estimar la concentración de cadmio y cobre mediante la concentración de otros metales, la caracterización del tipo de piedra superficial y el uso que se le da al suelo. El interés en este problema recae en que, además de tener ruido y no-linealidad presente en sus observaciones, las concentraciones de los diferentes metales están altamente correlacionadas, de modo que se desea modelar un sistema con salidas correlacionadas [66]. Cuando se tiene un sistema con múltiples salidas y éstas están correlacionadas, la capacidad de pronóstico aumenta cuando se tiene la posibilidad de compartir información entre las diferentes tareas, en comparación con la realización individual de cada tarea (la inferencia de cada salida es considerada una tarea) [2]. En resumen, los problemas estudiados en esta tesis han sido seleccionados por características como su complejidad, su no-linealidad o ausencia de ruido, y por su capacidad para aportar conclusiones importantes en base a los resultados de cada método de solución. Hasta donde nosotros sabemos, éste es el primer estudio sistemático que evalúa el desempeño de la inferencia Bayesiana para problemas retadores de regresión en donde se considera el modelo de mezcla infinita de Gaussianas.

1.3 Objetivos

El objetivo principal de esta tesis consiste en evaluar la capacidad de predicción que tienen diversos métodos Bayesianos sobre problemas de estudio que involucran altos niveles de ruido y no-linealidad conforme el tamaño del conjunto de entrenamiento disminuye. Al reducir el número de observaciones disponibles para la etapa de entrenamiento, el efecto del ruido y la no-linealidad tienden a volverse más dominantes [46], lo que afecta considerablemente la capacidad de pronóstico de las técnicas

basadas en la estadística frecuentista al aumentar el riesgo de entrenar un modelo con sobreajuste. Para evitar el sobreajuste se necesita acoplar al entrenamiento alguna metodología que regule la complejidad del modelo. Los métodos Bayesianos contienen de manera implícita un mecanismo que regula tal complejidad, por lo que se espera que sus pronósticos sean más asertivos que aquellos provenientes de métodos frecuentistas, por lo que se realiza una comparación con una técnica que ha mostrado resultados precisos en problemas donde otros métodos han fallado. Un objetivo colateral de esta tesis es la implementación computacional de los métodos de solución propuestos.

1.4 Estructura de la Tesis

El resto de este documento está conformado por seis capítulos. En el Capítulo 2 se presenta una introducción a la inferencia Bayesiana, comparándola con la estadística frecuentista clásica y detallando al mismo tiempo sus componentes. En el Capítulo 3 se describen los problemas de estudio considerados en esta tesis, los cuales son problemas ruidosos no-lineales cuya estimación mediante métodos clásicos requiere de un ajuste de complejidad laborioso, por lo que se aplican diversos métodos Bayesianos estadísticos y de aprendizaje automático, los cuales son desarrollados y descritos a profundidad en el Capítulo 4. En ese mismo capítulo se presenta el método *bootstrap*. Posteriormente, en el Capítulo 5 se analizan los resultados obtenidos al aplicar los métodos de solución a los problemas de estudio y se discute su impacto en este estudio. Finalmente, el Capítulo 6 contiene las conclusiones y observaciones que se han derivado de esta investigación. De manera adicional, en el Apéndice A se presenta una descripción de la implementación computacional de los métodos de solución Bayesianos, la cual sirve como guía para lograr su efectiva implementación computacional. La referencia proporcionada en este apéndice es particularmente útil para el modelo de mezcla infinita de Gaussianas, el cual no ha sido implementado en ninguna herramienta de modelación estadística.

Capítulo 2

Marco Teórico

La inferencia estadística es el proceso mediante el cual se realizan estimaciones acerca de un parámetro poblacional a partir de un conjunto de observaciones que provienen de un sistema afectado por variabilidad aleatoria [19]. Actualmente existen dos tipos de filosofías estadísticas que predominan en cuestiones de inferencia: la inferencia frecuentista y la Bayesiana, las cuales difieren en la interpretación del significado de probabilidad y de su aplicación en la estimación de parámetros [5]. A pesar de que la teoría Bayesiana fue desarrollada durante el siglo XIX, su aplicación presupone la resolución de complejas integrales que en muchas ocasiones no pueden resolverse analíticamente, lo que condujo a que se perdiera interés en el desarrollo de métodos Bayesianos y se optara por adoptar la filosofía frecuentista como base estadística. Es hasta nuestros tiempos cuando se tiene la capacidad de aplicar la inferencia Bayesiana en problemas reales, debido a que el avance tecnológico ha permitido el desarrollo de equipos de cómputo accesibles y capaces de desempeñar cálculos que hace algunos años eran imposibles de realizar, conduciendo al desarrollo de nuevas técnicas de aproximación numérica [21]. En la Sección 2.1 se introduce la inferencia estadística a partir de la metodología clásica frecuentista, en donde brevemente se discuten las desventajas que conlleva su uso en problemas reales, lo cual sirve de motivación para presentar el enfoque Bayesiano. Posteriormente, en la Sección 2.2 se introducen los métodos de Monte Carlo basados en cadenas de Markov que se utilizan como métodos de aproximación para la inferencia Bayesiana en esta tesis, como lo son el muestreo de Gibbs y el muestreo de Gibbs con paso

Metrópolis. Finalmente, se introduce el recocido simulado como una técnica de optimización global que permite tomar un camino diferente al muestreo aleatorio como aproximación.

2.1 Inferencia estadística Bayesiana

La teoría detrás de la inferencia estadística consiste en todas aquellas técnicas a través de las cuales se puede realizar inferencia sobre una variable aleatoria bajo estudio [19]. Hasta hace algunos años, los métodos que dominaban el área de la inferencia estadística estaban basados en el marco teórico de la estadística frecuentista, en donde se toma una muestra aleatoria que se supone proviene de una distribución probabilística con parámetros $\boldsymbol{\theta}$ [19]. En esta filosofía se hace la suposición de que tales parámetros tienen asociados valores desconocidos, pero fijos, por lo que no es posible asociar una distribución de probabilidad a ellos. La única probabilidad considerada en la estadística frecuentista es la distribución de la muestra aleatoria de tamaño N dados los parámetros $\boldsymbol{\theta}$, la cual explica cómo es que la muestra aleatoria observada fluctúa sobre todas las posibles muestras aleatorias dados los parámetros fijos $\boldsymbol{\theta}$ [5]. De esta forma, la probabilidad de observar una muestra determinada se interpreta como el límite de su frecuencia relativa en una gran cantidad de experimentos, de donde finalmente adquiere el nombre de estadística frecuentista [5].

El punto característico de la estadística frecuentista es que los parámetros $\boldsymbol{\theta}$ que componen un modelo tienen valores desconocidos, pero fijos. Una forma clásica de determinar estos parámetros es aproximarlos mediante estimadores puntuales, $\widehat{\boldsymbol{\theta}}$, los cuales se construyen utilizando las observaciones disponibles y por ende tienen una dependencia en la muestra aleatoria observada, de modo que son variables aleatorias cuya distribución de probabilidad corresponde con la distribución probabilística de la muestra [63]. Ahora bien, si la distribución muestral está centrada cerca del valor real (pero desconocido) de los parámetros y ésta no presenta demasiada dispersión, entonces los estadísticos pueden ser utilizados como estimadores de

los parámetros, recibiendo el nombre de estimadores puntuales. No se espera que un estimador puntual realice la estimación del parámetro poblacional sin error, sino que en realidad se espera que no se encuentre muy alejado de éste. No obstante, no es posible juzgar la precisión de los estimadores puntuales porque no se conoce el valor real de los parámetros, por lo que se utiliza un criterio basado en la distribución muestral de los estimadores y que es equivalente a la distribución de los estimadores sobre todas las posibles muestras aleatorias [19]. Aún así, es improbable que incluso el estimador insesgado más eficiente (i.e., el que tenga una menor varianza) estime el parámetro poblacional con exactitud. A pesar que la precisión del estimador puntual aumenta conforme se incrementa el tamaño muestral, no hay razón para suponer que la estimación puntual de una muestra dada sea exactamente igual al parámetro poblacional que se desea estimar [5]. Un procedimiento más adecuado consiste en determinar intervalos de estimación que tengan una probabilidad predeterminada de contener los parámetros desconocidos. Aunque los parámetros son considerados constantes, los extremos de tales intervalos son aleatorios porque dependen de la muestra aleatoria observada. Contradictoriamente, cuando la muestra es observada y los extremos son identificados no queda nada de aleatoriedad presente en la estimación, por lo que a estos intervalos se les conoce como intervalos de confianza. Se conoce de antemano que una determinada *proporción* de los intervalos construidos mediante muestras aleatorias contendrá los parámetros reales, pero no se puede concluir nada acerca de los intervalos de confianza específicos calculados con la muestra observada. Esto representa una clara desventaja del método frecuentista, ya que un parámetro está dentro del intervalo de confianza o simplemente no lo está, de manera que no es posible afirmar que existe una cierta *probabilidad* de que el parámetro se encuentre dentro del intervalo, situación que se deriva de la suposición inicial en el método frecuentista de asumir que el parámetro toma un valor fijo pero desconocido, impidiendo que se le pueda asociar una distribución de probabilidad.

Detrás de la inferencia frecuentista existen años de teoría, así como una gran cantidad de técnicas basadas en la distribución de las muestras aleatorias, de entre las cuales se distingue la estimación por máxima verosimilitud (MLE) como una técnica

para estimar los parámetros de un modelo estadístico. La estimación por máxima verosimilitud es un método estándar ampliamente reconocido en la actualidad [23] por tener un gran número de propiedades óptimas para estimación, como lo son la suficiencia (i.e., información completa sobre el parámetro de interés contenida en su estimador MLE), consistencia (i.e., el valor real del parámetro se obtiene asintóticamente conforme aumenta el tamaño muestral), eficiencia (i.e., la menor varianza posible del parámetro estimado se obtiene también asintóticamente), y la invariabilidad del modelo (i.e., el mismo estimador MLE se obtiene independientemente de la parametrización elegida), por lo que esta técnica es la columna vertebral del desarrollo de un gran número de métodos de inferencia estadística. Dado que diferentes valores para $\boldsymbol{\theta}$ implican diferentes distribuciones de probabilidad, el principio detrás de la estimación por máxima verosimilitud desarrollado por Fisher plantea que la distribución de probabilidad de interés debe ser aquella que maximiza la probabilidad de obtener la muestra observada [23], lo que a su vez implica que debe buscarse el conjunto de parámetros que maximicen la distribución de los parámetros dadas las observaciones, $p(\boldsymbol{\theta}|\mathcal{D})$, conocida en la literatura como función de verosimilitud, la cual será abordada más delante. Así, se intuye que la estimación por máxima verosimilitud tiene una dependencia en la muestra aleatoria observada. Para una descripción más profunda sobre la estimación por máxima verosimilitud, véase [23].

Un importante aspecto a considerar al utilizar la estimación por máxima verosimilitud es la complejidad del modelo sobre el que se hace el ajuste de parámetros. Si se tiene un modelo demasiado complejo, un pequeño cambio en el conjunto de observaciones puede causar un cambio radical en sus parámetros estimados, de modo que su varianza se incrementa [9]. Sin embargo, un modelo más complejo permite un mejor ajuste de los datos, de modo que el sesgo decrece [9]. A esta situación en que existe una relación inversa entre el sesgo y la varianza en función de la complejidad del modelo se le conoce como el dilema sesgo/varianza, y ocurre en todo sistema de aprendizaje automático y estadístico [20]. Para disminuir el sesgo, el modelo debe ser lo suficientemente flexible, ante el riesgo de tener una mayor varianza. Si por el contrario, la varianza se trata de mantener en sus niveles más bajos, se puede no

tener un buen ajuste de las observaciones, lo que conduce a un aumento en el sesgo. Así, el modelo óptimo es aquel que tiene el mejor equilibrio entre sesgo y varianza. Adicionalmente, la variabilidad concerniente a las observaciones disminuye conforme se aumenta el tamaño muestral [63], de modo que aumentar la cantidad de observaciones disminuye tanto el sesgo como la varianza. A la situación particular en que el modelo es demasiado general (o demasiado complejo) y aprende también el ruido presente en la muestra se le conoce como sobreajuste [12]. El riesgo de aprender un modelo con sobreajuste representa una clara desventaja de la filosofía frecuentista, debido a que un modelo que presenta sobreajuste tiene una baja capacidad de generalización[1], como se muestra en la Figura 2.1 sobre observaciones fuera de la muestra. Para sobrellevar esta problemática existen un gran número de técnicas que ajustan la complejidad del modelo, como lo son la validación cruzada [32], la regularización [47], la minimización del riesgo estructural [56], la longitud de descripción mínima [68], los métodos *bootstrap* [16] y los métodos Bayesianos [5], por mencionar algunos. De éstas, la validación cruzada es la más empleada para labores de control de complejidad [12], siendo la única que no hace suposiciones de antemano acerca del modelo. No obstante, tiene como desventaja el que se deba destinar una proporción de las observaciones para evaluar el desempeño del modelo, lo que reduce el tamaño de la muestra disponible para entrenar el modelo. De esta forma, la validación cruzada es la técnica más efectiva cuando existe un conjunto grande de observaciones [12], mientras que el resto de los procedimientos se vuelven útiles cuando el conjunto de observaciones es pequeño. Una variación de la validación cruzada que ha sido desarrollada para aplicarse cuando se tienen pocas observaciones es la validación cruzada tipo *leave-one-out* [17], en donde se evalúa el error de generalización de la n-ésima observación utilizando las $(N-1)$ observaciones restantes para estimar los parámetros del modelo, variando n de tal forma que se evalúe la capacidad de pronóstico para cada observación en la muestra. El error de generalización estimado corresponde con la media de los N errores. Sin embargo, dado que el estimador

[1]En esta tesis se utilizan indistintamente generalización, pronóstico y predicción, para indicar la inferencia en una muestra no observada anteriormente.

requiere que el entrenamiento por máxima verosimilitud sea efectuado N veces, este procedimiento es computacionalmente costoso cuando se tienen conjuntos grandes de entrenamiento, además de dar como resultado un estimador con varianza relativamente alta [12], que puede conducir a conclusiones erróneas acerca del nivel de sobreajuste. Otro método capaz de estimar el error de generalización de un modelo basado en remuestreo es el *bootstrap*, el cual parece funcionar mejor que la validación cruzada en muchos casos, resultando en estimadores con una menor varianza que aquellos de la validación cruzada [16]. El *bootstrap* es, en esencia, una implementación computacional de una estimación paramétrica de máxima verosimilitud [24], el cual se introduce en la Sección 4.4.

2.1.1 El teorema de Bayes

De entre los métodos descritos para prevenir el sobreajuste, los métodos Bayesianos han ganado una gran popularidad en los últimos años. Las ventajas de aplicar la estadística Bayesiana involucran (i) una estimación más intuitiva y significativa, al utilizar la teoría de probabilidad para denotar el grado de incertidumbre que existe en la estimación de un parámetro [42], (ii) la capacidad de incorporar información previa disponible antes de analizar una muestra [5], (iii) la prevención del sobreajuste, debido a que la inferencia Bayesiana contiene de forma implícita el principio de la navaja de Ockham y selecciona automáticamente el modelo con la mayor probabilidad posterior [28], de modo que (iv) no necesita un conjunto de validación, por lo que toda la muestra puede utilizarse en el entrenamiento [12]. No obstante, la inferencia Bayesiana sufre de un problema que ha sido ampliamente criticado: el conocimiento a-priori, en donde la creencia personal acerca de la distribución de los parámetros a estimar se involucra en los cálculos, y la influencia de ésta adiciona un cierto grado de subjetividad al proceso [1].

La tendencia actual es distinguir entre los métodos clásicos de estimación, los cuales tienen un fondo frecuentista que construye su inferencia mediante información obtenida de una muestra seleccionada aleatoriamente de la población, y los métodos

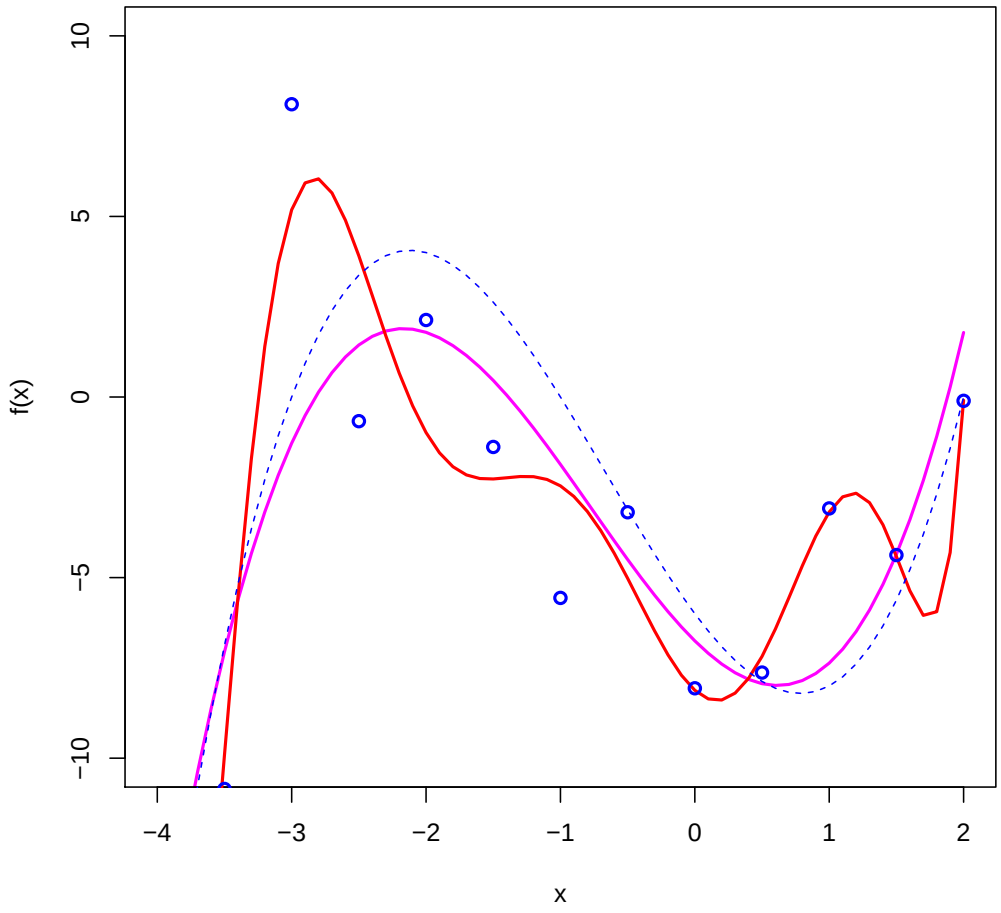


Figura 2.1: Ilustración del sobreajuste. La línea azul punteada representa la función cúbica $f(x) = (x+3)(x+1)(x-2)$, de donde se muestrean aleatoriamente los puntos mostrados como círculos con un ruido aditivo, tal que $f(x) = (x+3)(x+1)(x-2) + \mathcal{N}(0,2)$. En magenta se muestra un polinomio de tercer grado ajustado sobre las observaciones tomadas, la cual intenta aproximar la función original ignorando el ruido blanco. Por el contrario, en rojo se presenta un polinomio de sexto grado que, al ser más complejo, intenta emular cada observación tomada, aprendiendo también el ruido blanco. Los coeficientes de ambos polinomios fueron calculados mediante mínimos cuadrados.

basados en la filosofía Bayesiana [5], en donde se utiliza un conocimiento subjetivo a priori acerca de la distribución de probabilidad de los parámetros en conjunto con la información que aporta la muestra observada. La diferencia más notable entre los métodos frecuentistas y los métodos Bayesianos es que en los segundos los parámetros del modelo son considerados variables aleatorias, lo que implica que tienen una distribución de probabilidad asociada que representa la incertidumbre que se tiene acerca de su valor [63].

Teorema 2.1 (Teorema de Bayes) *Sea* $\{A_1, A_2, \ldots, A_i, \ldots, A_n\}$ *un conjunto de sucesos mutuamente excluyentes y exhaustivos, tales que la probabilidad de cada uno de ellos es distinta de cero. Sea* B *un suceso cualquiera del que se conocen las probabilidades condicionales* $P(B|A_i)$*. Entonces, la probabilidad* $P(A_i|B)$ *viene dada por la expresión:*

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)}, \tag{2.1}$$

en donde $P(A_i)$ *son las probabilidades a priori,* $P(B|A_i)$ *es la probabilidad de* B *en la hipótesis* A_i *y* $P(A_i|B)$ *son las probabilidades posteriores.*

El teorema de Bayes (Teorema 2.1) sustenta el marco teórico detrás de la inferencia Bayesiana, e indica una forma de evaluar la probabilidad de un suceso A una vez que se conoce que el sucedo B ha ocurrido, siendo A y B sucesos con una dependencia mutua. Si el evento B representa observar la muestra $\mathcal{D}$ y el suceso A denota que los parámetros del modelo sean $\boldsymbol{\theta}$, entonces (2.1) puede reescribirse como:

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})}, \tag{2.2}$$

en donde $p(\boldsymbol{\theta})$ corresponde con las distribuciones a priori de los parámetros antes de que se observe la muestra, $p(\boldsymbol{\theta}|\mathcal{D})$ es la distribución posterior de interés que indica la

distribución de los parámetros dadas las observaciones, expresando la incertidumbre que se tiene acerca de los parámetros después de tomar en cuenta tanto las observaciones como la distribución a priori, y finalmente, $p(\mathcal{D}|\boldsymbol{\theta})$ es la probabilidad de observar la muestra $\mathcal{D}$ dado que se tienen los parámetros $\boldsymbol{\theta}$, es decir, la evidencia que aportan las observaciones en favor de los parámetros. Este último término evalúa la verosimilitud de los parámetros del modelo dadas las observaciones, y es una distribución importante en el aprendizaje estadístico debido a que relaciona todas las variables por medio de un modelo completamente probabilístico. Para encontrar el conjunto de parámetros más probables, $\langle\boldsymbol{\theta}\rangle$, se realiza una marginalización de la distribución posterior con respecto a los parámetros:

$$\langle\boldsymbol{\theta}\rangle = \int \boldsymbol{\theta}\ p(\boldsymbol{\theta}|\mathcal{D})\ d\boldsymbol{\theta}. \tag{2.3}$$

La integración de la función de verosimilitud marginal con respecto a los parámetros es lo que distingue la filosofía Bayesiana de otros esquemas basados en optimización, y es el término que introduce de manera automática un equilibrio entre el ajuste del modelo y su complejidad, previniendo el sobreajuste [5]. Por su parte, el denominador de (2.2) define la verosimilitud marginal de las observaciones:

$$p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{\theta})\ p(\boldsymbol{\theta})\ d\boldsymbol{\theta}. \tag{2.4}$$

Esta verosimilitud marginal, también conocida como distribución predictiva a priori de las observaciones, indica la forma que se espera tengan las observaciones antes de que éstas sean observadas, y depende únicamente de las distribuciones a priori de los parámetros y de la verosimilitud del modelo, siendo independiente de los parámetros $\boldsymbol{\theta}$, por lo que se trata de un factor constante que normaliza el numerador $p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ para que sea una distribución de probabilidad propia. De este forma, la distribución posterior de interés es proporcional al producto de la verosimilitud de los parámetros y su distribución a priori:

$$p(\boldsymbol{\theta}|\mathcal{D}) \propto p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{2.5}$$

Esta proporcionalidad es importante debido a que en algunos casos es posible evaluar $p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ e inspeccionar la forma de la distribución posterior no-normalizada. Si esta distribución es equivalente a alguna distribución conocida, entonces no es necesario evaluar explícitamente la constante de normalización en (2.4) [28]. Para que esto ocurra, la verosimilitud y las distribuciones a priori deben complementarse de tal forma que la distribución posterior no-normalizada tenga una forma conocida.

2.1.2 Componentes de la inferencia Bayesiana

La distribución posterior de los parámetros dadas las observaciones, $p(\boldsymbol{\theta}|\mathcal{D})$, depende de dos componentes [5]. El primero de ellos corresponde con la verosimilitud de los parámetros dada la evidencia. La función de verosimilitud, $p(\mathcal{D}|\boldsymbol{\theta})$, contiene la información proporcionada por las observaciones e indica la probabilidad de tener los parámetros $\boldsymbol{\theta}$ dado que se tienen las observaciones $\mathcal{D}$, y se evalúa mediante la distribución de probabilidad de cada observación dados los parámetros. Suponiendo que $\mathcal{D} = (y_1, \ldots, y_N)^T$, entonces:

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{n=1}^{N} p(y_n|\boldsymbol{\theta}). \tag{2.6}$$

La verosimilitud representa la evidencia que las observaciones aportan acerca de los parámetros [42]. Por otro lado, el segundo componente de la inferencia Bayesiana corresponde con las distribuciones a priori de los parámetros, $p(\boldsymbol{\theta})$, las cuales son distribuciones de probabilidad en el espacio de los parámetros y representan la incertidumbre que se tiene acerca de éstos antes de tomar en cuenta las observaciones. Los parámetros de una distribución a priori son llamados hiperparámetros, para distinguirlos de los parámetros del modelo, $\boldsymbol{\theta}$ [42]. Existen dos tipos de distribuciones a priori: las distribuciones informativas y las no-informativas.

Cuando existe información previa acerca de los parámetros ésta debe incluirse en la distribución a priori [5]. Por ejemplo, esto ocurre cuando el modelo bajo estudio es similar a un modelo previo cuyos parámetros serán actualizados mediante un nuevo conjunto de observaciones. Haciendo uso de la distribución posterior del modelo previo como distribución a priori del modelo bajo estudio, el modelo no comienza basándose únicamente en la muestra obtenida, sino que estima los nuevos parámetros mediante un efecto acumulativo de las observaciones anteriores y las actuales. Por el contrario, cuando no existe información previa acerca de los parámetros, situación que se presenta más comúnmente, se hace uso de una distribución vaga o no-informativa para minimizar el impacto que tiene seleccionar la distribución a priori. Esto implica que la distribución a priori tendrá un menor impacto en la distribución posterior del modelo, mientras que la información proporcionada por las observaciones tendrá mayor peso. La distribución a priori no-informativa más sencilla corresponde con la distribución uniforme, también llamada distribución a priori plana, debido a su distribución probabilística constante:

$$p(\boldsymbol{\theta}) \sim \mathcal{U}(-\infty, \infty), \tag{2.7}$$

en donde $\boldsymbol{\theta}$ está uniformemente distribuida del infinito negativo al infinito positivo. Sin embargo, a pesar de que la distribución uniforme permite que la distribución posterior sea afectada exclusivamente por las observaciones, ésta será impropia y no integrará a uno, dado que la integral de la distribución en (2.7) es infinita, violando uno de los axiomas de la teoría de probabilidad [37]. Así, una distribución impropia ocurre cuando:

$$\int p(\boldsymbol{\theta})\ d\boldsymbol{\theta} = \infty. \tag{2.8}$$

Para un análisis más extenso sobre distribuciones a priori, véase [67].

2.1.3 Predicciones en la inferencia Bayesiana

Idealmente, en un enfoque Bayesiano se definen la función de verosimilitud y las distribuciones a priori de los parámetros, para posteriormente utilizar el teorema de Bayes desarrollado en (2.2) y evaluar la distribución posterior de los parámetros dadas las observaciones:

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})}. \tag{2.9}$$

Esta distribución probabilística es la base para realizar inferencia de acuerdo a la filosofía Bayesiana [5]. Supóngase que se tiene una observación y_f que se desea inferir y que no ha sido utilizada para entrenar el modelo. La inferencia toma lugar mediante la verosimilitud de la nueva observación promediada sobre la distribución posterior $p(\boldsymbol{\theta}|\mathcal{D})$:

$$p(y_f|\mathcal{D}) = \int p(y_f|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta}. \tag{2.10}$$

Sin embargo, cuando el numerador de (2.9) no puede ser evaluado analíticamente, entonces tampoco es posible evaluar analíticamente la integral en (2.10) [37], de modo que es necesario implementar una aproximación para resolver esta integral. Entre las aproximaciones más comúnmente aplicadas en la literatura se encuentran el método computacional de aproximación Bayesiana [49], el método Bayes variacional [4], las aproximaciones de Laplace [60], los métodos Monte Carlo basados en cadenas de Markov [21] y la estimación de la distribución posterior máxima [37]. Los dos últimos métodos se introducen en esta sección y son utilizados en esta tesis como técnicas de aproximación.

Estimación de la distribución posterior máxima. La estimación por medio de la distribución posterior máxima implica aproximar la integral en (2.10) haciendo uso de los valores más probables de los parámetros. Esto es posible debido a que

cuando la distribución condicional $p(\boldsymbol{\theta}|\mathcal{D})$ se encuentra en su punto máximo, el estimador posterior de $\boldsymbol{\theta}$ corresponde también con un punto máximo [37]. Sea $\boldsymbol{\theta}_{MP}$ el conjunto óptimo de parámetros, la aproximación que se realiza es:

$$p(y_f|\mathcal{D}) \simeq p(y_f|\mathcal{D}, \boldsymbol{\theta}_{MP}). \tag{2.11}$$

El teorema de Bayes indica que cuando no se tiene suficiente información a priori acerca de los parámetros $\boldsymbol{\theta}$, el punto máximo de la distribución posterior $p(\boldsymbol{\theta}|\mathcal{D})$ corresponde con el punto máximo de la verosimilitud, $p(\mathcal{D}|\boldsymbol{\theta})$. De esta forma, para encontrar el conjunto de parámetros más probable basta con aplicar un algoritmo de optimización para maximizar $p(\mathcal{D}|\boldsymbol{\theta})$ con respecto a los parámetros. No obstante, esta aproximación comparte algunas de las desventajas que exhibe la estimación por máxima verosimilitud, como lo son la posible atracción por óptimos locales deficientes y la necesidad de una exploración adecuada en funciones no-diferenciables. Un método útil de optimización global que intenta reducir estas dificultades es el recocido simulado, el cual se introduce en la Sección 2.3.

Métodos Monte Carlo. El segundo método de aproximación corresponde con los métodos Monte Carlo [21]. Supóngase que se tiene la capacidad de simular un conjunto de M muestras independientes tomadas de la distribución posterior de los parámetros en (2.9). Sea $\{\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^M\}$ el conjunto de tales muestras, el estimador Monte Carlo para la integral en (2.10) corresponde con la media de este conjunto, de modo que:

$$p(y_f|\mathcal{D}) \simeq \frac{1}{M} \sum_{m=1}^{M} p(y_f|\boldsymbol{\theta}^m). \tag{2.12}$$

El método Monte Carlo es un algoritmo efectivo para resolver numéricamente integrales complejas o analíticamente intratables, como la integral en (2.10), cuando es posible tomar muestras independientes de la distribución posterior exacta, lo cual no siempre ocurre. Afortunadamente, existen variantes de este método que permiten

generar tales muestras mediante la construcción de una cadena de Markov. Estos son los métodos Monte Carlo basados en cadenas de Markov (MCMC), los cuales se introducen en la siguiente sección.

2.2 Métodos Monte Carlo basados en Cadenas de Markov

Los métodos MCMC son una clase de algoritmos iterativos que generan muestras provenientes de una distribución de probabilidad mediante la construcción de una cadena de Markov que converge a la distribución deseada como su distribución de equilibrio [21], por lo que son aplicados en la inferencia Bayesiana para muestrear de la distribución posterior de los parámetros, $p(\boldsymbol{\theta}|\mathcal{D})$. Si estas muestras provienen de una sucesión de variables aleatorias independientes e idénticamente distribuidas, entonces el estimador Monte Carlo en (2.12) es un estimador consistente del verdadero valor de la integral conforme $M \to \infty$, de acuerdo a la Ley de los Grandes Números. Sin embargo, construyendo una cadena de Markov se obtienen muestras que son ligeramente dependientes un paso atrás en el tiempo, de acuerdo a la Definición 2.2, pero el teorema ergódico (Teorema 2.3) permite ignorar la dependencia entre muestras provenientes de una cadena de Markov cuando $M \to \infty$ [7]. La habilidad de los métodos MCMC para generar muestras ligeramente dependientes es particularmente útil cuando no se conoce la constante de normalización en el teorema de Bayes. Para una descripción clara y concisa acerca del teorema ergódico y las propiedades de las cadenas de Markov, véase [7].

Definición 2.2 *Una cadena de Markov es un proceso estocástico en donde los estados futuros dependen únicamente del estado actual.*

Teorema 2.3 (Teorema ergódico) *Si* $\{w^1, \ldots, w^m\}$ *es un conjunto de* M *muestras obtenidas de una cadena de Markov que es aperiódica, irreducible y recurrente, entonces tal cadena de Markov es ergódica. Esto implica que:*

$$\frac{1}{M}\sum_{m=1}^{M} p(y_f|\boldsymbol{\theta}^m) \rightarrow p(y_f|\mathcal{D}), \tag{2.13}$$

conforme $M \rightarrow \infty$.

Un método MCMC es entonces una clase de algoritmos en que se simulan una serie de muestras (ligeramente dependientes) provenientes de una distribución posterior mediante la construcción de una cadena de Markov. Tales muestras son utilizadas en el método Monte Carlo para evaluar numéricamente las integrales complejas que ocurren frecuentemente en la inferencia Bayesiana, produciendo estimaciones de las propiedades de interés de la distribución posterior bajo estudio. En inferencia Bayesiana hay dos algoritmos que suelen utilizarse con frecuencia: el muestreo de Gibbs (Sección 2.2.1) y el muestreo de Gibbs con paso Metrópolis (Sección 2.2.2).

2.2.1 El muestreo de Gibbs

Uno de los métodos más atractivos para implementar un algoritmo MCMC es el muestreo de Gibbs, el cual tiene sus raíces en la mecánica estadística [21]. Supóngase que $\boldsymbol{\theta}$ está conformado por q parámetros de interés, de modo que $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_q)^T$. La distribución posterior de los parámetros dadas las observaciones, $p(\boldsymbol{\theta}|\mathcal{D})$, puede ser de gran dimensión y difícil de evaluar. Si es posible definir la distribución probabilística univariada para el k-ésimo elemento de $\boldsymbol{\theta}$ condicionado tanto en el resto de los parámetros como en las observaciones, tal que:

$$p(\theta_k|\theta_1, \ldots, \theta_{k-1}, \theta_{k+1}, \ldots, \theta_q, \mathcal{D}) \qquad \forall k, \tag{2.14}$$

entonces esta distribución es más sencilla de simular que la distribución posterior completa, al tener generalmente formas más simples [64]. El muestreo de Gibbs consiste en la ideología de que es posible construir una cadena de Markov para la distribución posterior simulando secuencialmente el k-ésimo parámetro mediante su

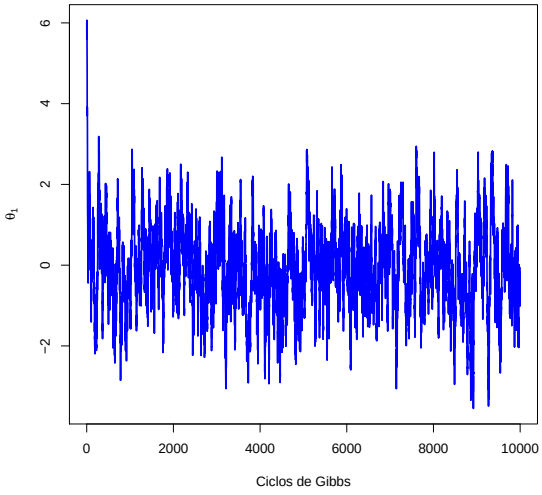
distribución condicional (ec. 2.14), en vez de generar una sola muestra de q componentes de la distribución posterior completa. Al muestreo de todos los parámetros en una iteración se le denomina un ciclo de Gibbs. Bajo condiciones generales, las muestras generadas por este método convergen a la distribución objetivo de interés [21].

El muestreo comienza con algún valor inicial para todas las variables con excepción de θ_1, cuyo valor se muestrea a partir de la distribución condicional $p(\theta_1|\theta_2, \ldots, \theta_q,\ \mathcal{D})$. Posteriormente se genera un nuevo valor para θ_2 simulando de su distribución condicional $p(\theta_2|\theta_1, \theta_3, \ldots, \theta_q, \mathcal{D})$, luego se muestrea θ_3 mediante $p(\theta_3|\theta_1, \theta_2, \theta_4, \ldots, \theta_q,\ \mathcal{D})$ y así sucesivamente hasta muestrear θ_q, consolidando el primer ciclo de Gibbs. El muestreo continúa iniciando de nuevo con θ_1, formando un nuevo ciclo. La secuencia de Gibbs converge a una distribución de equilibrio (la distribución posterior de interés) independientemente de los valores iniciales después de pasar por una etapa inicial transitoria. Por tanto, sólo un subconjunto de tamaño M del total de las muestras contiene observaciones simuladas de la distribución posterior.

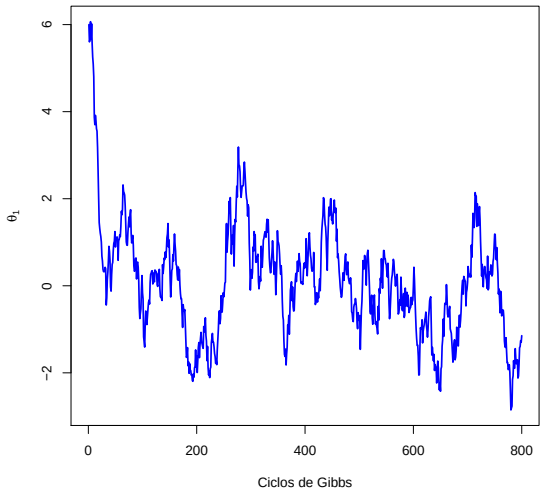
Dependencia del método en los valores iniciales: burn-in. Un punto clave en la implementación de un método MCMC consiste en determinar el número de ciclos que ocurren antes que la cadena se aproxime a su distribución estacionaria, donde el estado actual ha dejado de ser dependiente de las condiciones iniciales del muestreo. Al período transitorio que toma lugar antes que la cadena alcance la estacionalidad se le conoce en la literatura como *burn-in* [21]. De manera típica, en esta tesis se eliminan una cantidad que oscila entre los 1,000 y 5,000 ciclos iniciales y se aplica una prueba de convergencia para asegurarse que se ha alcanzado la estacionalidad. A pesar que los algoritmos MCMC garantizan la estacionalidad [64], esta garantía no determina el tiempo que tardarán en alcanzarla. Por ejemplo, una pobre selección de los puntos iniciales del muestreo puede incrementar dramáticamente la cantidad de ciclos que la cadena pasa en la etapa *burn-in*, por lo que el desarrollo de metodologías que sean capaces de estimar puntos de inicio óptimos es un área

de interés actual. En esta tesis se emplean reglas básicas para la estimación de la longitud del estado transitorio, como lo son los gráficos de series temporales. La Figura 2.2a muestra la serie temporal para una de dos variables provenientes de una distribución Gaussiana bivariada, mientras que en la Figura 2.2b puede observarse el efecto que tienen los valores iniciales en la cadena de Markov hasta los 200 ciclos. A partir de los 200 ciclos, la gráfica parece oscilar alrededor de un valor medio, lo que a su vez parece indicar que se ha alcanzado la estacionalidad, como se observa en la Figura 2.2c. A pesar de que es posible afirmar si una cadena se encuentra fuera de equilibrio, no es posible determinar cuando éste ha sido alcanzado debido a que existen puntos que producen estados estacionarios metaestables.

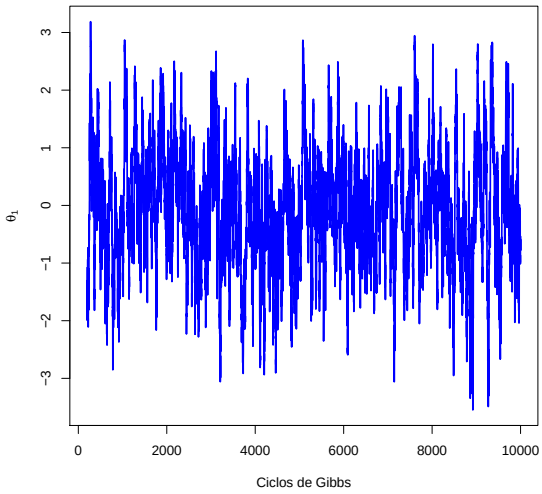
Autocorrelación en las muestras: el factor de inflación. Cuando se han realizado las pruebas pertinentes y se concluye (con algún nivel de confianza) que la cadena ha alcanzado la estacionalidad, se descartan las muestras concernientes a la etapa *burn-in*, de modo que el conjunto restante está formado por muestras provenientes de la distribución de equilibrio deseada. Sin embargo, estas muestras son ligeramente dependientes en el tiempo, por lo que se espera que las muestras generadas en ciclos de Gibbs subsecuentes tengan una correlación positiva, de acuerdo a la definición de un proceso de Markov (Definición 2.2). Un resultado importante en la teoría del análisis de series de tiempo indica que si la serie proviene de un proceso estacionario y correlacionado, entonces las muestras correlacionadas aún proporcionan una estimación insesgada de la distribución, asumiendo que el tamaño muestral es suficientemente grande [10]. A pesar que el teorema ergódico permite extraer la misma conclusión, la teoría de series de tiempo permite además estimar el tamaño de la muestra mediante el cual se tiene la capacidad de realizar estimaciones insesgadas. Considérese una secuencia de muestras de longitud M, es decir, $\{\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^M\}$. La correlación puede estar presente entre muestras subsecuentes, tal que $p(\theta_k^t, \theta_k^{t+1}) \neq 0$, o de forma general, entre muestras más distantes, es decir, $\rho(\theta_k^t, \theta_k^{t+j}) \neq 0$. La función de autocorrelación para la serie temporal de un proceso es simplemente la correlación de dicho proceso con una versión desplazada en el tiempo de si misma, por lo



(a) Muestreo de Gibbs.



(b) Efecto de los valores iniciales.



(c) Presunta estacionalidad alcanzada.

Figura 2.2: Gráfica de la serie temporal de un muestreo de Gibbs para una distribución normal bivariada con media $\mu = 0$, varianza marginal $\sigma^2 = 1$ para cada variable y coeficiente de correlación $\rho = 0.98$. El muestreo completo se muestra en (a), iniciando con ambas variables tomando el valor de 6. En (b) se observa la dependencia que tiene el valor inicial durante los primeros 200 ciclos del muestreo, etapa que comprende el *burn-in*. En (c) se presenta la cadena de Markov sin las muestras pertenecientes al *burn-in*, mostrando lo que parece ser una estacionalidad.

que mide la autocorrelación entre θ_k^t y θ_k^{t+p} con un retraso p. La autocorrelación de orden j para el k-ésimo parámetro al tiempo t puede estimarse mediante:

$$\widehat{\rho}_k^j = \frac{\mathrm{Cov}(\theta_k^t, \theta_k^{t+j})}{\mathrm{Var}(\theta_k^t)} = \frac{\sum_{t=1}^{M-j} (\theta_k^t - \widehat{\theta}_k)(\theta_k^{t-j} - \widehat{\theta}_k)}{\sum_{t=1}^{M-j} (\theta_k^t - \widehat{\theta}_k)^2} \qquad \forall j, k. \tag{2.15}$$

en donde:

$$\widehat{\theta}_k = \frac{1}{M} \sum_{t=1}^{M} \theta_k^t \qquad \forall k. \tag{2.16}$$

Suponiendo que las muestras son independientes, el estimador Monte Carlo en (2.12) necesita un mínimo de M_I muestras para estimar insesgadamente los parámetros en $\boldsymbol{\theta}$ [10]. Si las muestras no son independientes, una estimación del tamaño que debe tener la muestra proviene de la teoría de los procesos autoregresivos de primer orden, los cuales tienen una estructura similar a la caminata aleatoria:

$$\theta_k^t = \mu + \alpha(\theta_k^{t-1} - \mu) + \epsilon \qquad \forall k, \tag{2.17}$$

en donde ϵ corresponde con un ruido blanco, tal que $\epsilon \sim \mathcal{N}(0, \sigma^2)$, y α corresponde con la autocorrelación de primer orden para el k-ésimo parámetro, ρ_k, según (2.15). Bajo este proceso, el valor esperado de $\widehat{\theta}$ es $\langle \widehat{\theta} \rangle = \mu$, con desviación estándar:

$$\mathrm{S}(\widehat{\theta}_k) = \frac{\sigma}{\sqrt{n}} \sqrt{\frac{1 + \rho_k}{1 - \rho_k}} \qquad \forall k. \tag{2.18}$$

El primer término en (2.18) es la desviación estándar del ruido blanco, mientras que $\sqrt{(1 + \rho_k)/(1 - \rho_k)}$ es el factor de inflación del tamaño de muestra (SSIF), el cual muestra la forma en que la autocorrelación aumenta la varianza muestral. Por ejemplo, para $\rho_k = 0.5$, 0.75, 0.95 y 0.99, el SSIF asociado es 3, 7, 39 y 199,

respectivamente. De esta forma, con una autocorrelación de 0.95 se necesitan aproximadamente 40 veces el tamaño muestral de M_I para obtener la misma precisión que una muestra independiente.

2.2.2 El muestreo de Gibbs con paso Metrópolis

El muestreo de Gibbs es una técnica efectiva para generar muestras provenientes de una distribución cuando ésta no es de forma estándar o cuando es difícil muestrear directamente de ella [64]. Sin embargo, uno de los requerimientos del muestreo de Gibbs es que la distribución probabilística de cada parámetro condicionada en el resto de ellos pueda ser desarrollada analíticamente. Esto no es siempre posible, de manera que regularmente se opta por incorporar un algoritmo Metrópolis, como la caminata aleatoria, para simular de cada una de estas distribuciones condicionales [13].

El algoritmo Metrópolis. El algoritmo Metrópolis es otro método que tiene sus raíces en la mecánica estadística, y sobresale por tener la capacidad para muestrear de una distribución de probabilidad con el único requisito de que exista alguna función proporcional a la densidad de ésta que sea calculable [64]. Esto es particularmente útil en inferencia Bayesiana, dado que este algoritmo permite generar muestras de la distribución normalizada sin la necesidad de evaluar la constante de normalización [13], que en ocasiones resulta extremadamente difícil de evaluar. El algoritmo Metrópolis está basado en una analogía del equilibrio de los sistemas físicos cuyas configuraciones tienen probabilidad proporcional al factor de Boltzmann:

$$P(A) = \exp \frac{-E_A}{T}, \tag{2.19}$$

donde A es una configuración del sistema bajo estudio, T es una constante llamada *temperatura* y E representa la *energía* del sistema cuando se encuentra en su configuración A. Ahora bien, supóngase que se desea muestrear de una distribución $p(\boldsymbol{\theta})$,

tal que:

$$p(\boldsymbol{\theta}) = \frac{f(\boldsymbol{\theta})}{C}, \tag{2.20}$$

donde C representa una constante de normalización. El algoritmo Metrópolis inicializa con un valor θ_k^0 que satisface $f(\theta_k^0) > 0$. Posteriormente, y utilizando el valor actual para θ_k, se muestrea un estado candidato θ_k^* mediante una distribución de salto $q(\theta_k^t, \theta_k^{t+1})$, que representa la probabilidad de retornar un valor θ_k^{t+1} dado un valor previo θ_k^t. A esta distribución se le conoce en la literatura como distribución generadora de candidatos, y su única restricción es que sea simétrica [64], de modo que $q(\theta_k^t, \theta_k^{t+1}) = q(\theta_k^{t+1}, \theta_k^t)$. Dado el estado candidato θ_k^*, se evalúa la diferencia entre la *energía* en el estado actual, θ_k^t, y aquella en el estado candidato:

$$\alpha_k = \frac{\exp\left(-\dfrac{p(\theta_k^*, \boldsymbol{\theta}'^k)}{T}\right)}{\exp\left(-\dfrac{p(\theta_k^t, \boldsymbol{\theta}'^k)}{T}\right)} = \frac{\exp\left(-\dfrac{f(\theta_k^*, \boldsymbol{\theta}'^k)}{T}\right)}{\exp\left(-\dfrac{f(\theta_k^t, \boldsymbol{\theta}'^k)}{T}\right)} \tag{2.21}$$

$$= \exp\left(-\frac{f(\theta_k^*, \boldsymbol{\theta}'^k) - f(\theta_k^t, \boldsymbol{\theta}'^k)}{T}\right) \qquad \forall k, \tag{2.22}$$

donde $\boldsymbol{\theta}'^k$ indica todos los parámetros en $\boldsymbol{\theta}$ con excepción de θ_k. Debido a que se está considerando la razón de $p(\boldsymbol{\theta})$ para dos valores diferentes, la constante de normalización se cancela. Si el salto disminuye la energía del sistema ($\alpha_k > 1$), situación favorable de acuerdo al factor de Boltzmann, entonces se acepta el estado candidato como actual y se genera un nuevo candidato. Si, por el contrario, el salto incrementa la energía, entonces éste se acepta con probabilidad α_k, o se rechaza y se genera un nuevo candidato. De esta forma, el algoritmo Metrópolis consiste en evaluar:

$$\alpha_k = \text{mín}\left(1, \exp\left(-\frac{f(\theta_k^*, \boldsymbol{\theta}'^k) - f(\theta_k^t, \boldsymbol{\theta}'^k)}{T}\right)\right) \qquad \forall k, \tag{2.23}$$

y aceptar el estado candidato con probabilidad α_k. Este procedimiento genera una cadena de Markov, al estar el candidato en función del estado actual del sistema.

Acoplamiento al muestreo de Gibbs. La incorporación del algoritmo Metrópolis al muestreo de Gibbs es directa: en vez de muestrear cada variable mediante su distribución condicional, se da un paso Metrópolis mediante un mecanismo de perturbación y se acepta o rechaza. Sea θ_k^t el valor actual de θ_k en la simulación, el proceso de perturbación mediante el cual se generara un nuevo candidato es:

$$\theta_k^* = \theta_k^t + c_k Z \qquad \forall k, \tag{2.24}$$

donde Z es una variable con distribución normal estándar y c_k es un parámetro fijo de escala. El punto candidato θ_k^* se acepta con probabilidad:

$$\beta_k = \text{mín}\left(1, \exp\left(-\frac{f(\theta_k^*, \boldsymbol{\theta}'^k) - f(\theta_k^t, \boldsymbol{\theta}'^k)}{T}\right)\right) \qquad \forall k. \tag{2.25}$$

De otro modo, $\theta_k^{t+1} = \theta_k^t$. El parámetro c_k es ajustado de tal forma que la tasa de aceptación se encuentre dentro del intervalo $[0.3, 0.7]$, lo que corresponde con una aceptación entre el 30 % y el 70 % de los candidatos.

2.3 El método del recocido simulado

El recocido simulado es un algoritmo estocástico de optimización global desarrollado como un método para encontrar el máximo en funciones complejas multimodales, en donde los métodos clásicos basados en gradientes pueden quedar atrapados en un óptimo local de baja calidad [31]. La idea detrás del recocido simulado es que al iniciar la exploración del espacio de soluciones se tiene una probabilidad razonablemente alta de descender una colina (i.e., de aceptar un punto que disminuye el valor de la función objetivo) para realizar una mejor búsqueda del espacio de soluciones. Sin embargo, tal probabilidad decrece conforme el proceso continúa, de modo

que la exploración se da en pasos que mejoran la solución actual durante el final del proceso. La analogía en que se origina este método corresponde con el proceso físico de enfriamiento después del recocido de un cristal. Inicialmente existe mucho movimiento como resultado de las altas temperaturas, mientras que conforme avanza el proceso la libertad de movimiento se reduce. Algorítmicamente hablando, el recocido simulado es una técnica muy similar al muestreo Metrópolis, con la diferencia que esta vez la temperatura no es un factor constante, sino que está en función del avance del proceso, de modo que la probabilidad α_{SA} de dar un paso al punto candidato está dada por:

$$\alpha_{SA} = \text{mín}\left(1, \exp\left(-\frac{f(\theta_k^*, \boldsymbol{\theta}'^k) - f(\theta_k^t, \boldsymbol{\theta}'^k)}{T(t)}\right)\right), \tag{2.26}$$

en donde la función $T(t)$ recibe el nombre de programa de enfriamiento. De manera típica, una función con decrecimiento geométrico es utilizada. Por ejemplo, iniciando con una temperatura T_o y permitiendo el *enfriamiento* hasta T_f en k pasos:

$$T(t) = T_o\left(\frac{T_f}{T_o}\right)^{t/k}. \tag{2.27}$$

De forma más general, si se realiza un *enfriamiento* hasta la temperatura T_f al tiempo k, para luego mantener esta temperatura constante durante el resto del proceso, entonces:

$$T(t) = \text{máx}\left(\left(T_o\frac{T_f}{T_o}\right)^{t/k}, 1\right). \tag{2.28}$$

Como puede observarse, el algoritmo Metrópolis es un caso especial del recocido simulado en que la temperatura se mantiene fija. Por otro lado, la temperatura inicial del programa de enfriamiento debe seleccionarse cuidadosamente. Si ésta es muy alta, el método se aproxima a una búsqueda aleatoria, mientras que cuando la temperatura tiende a cero, el algoritmo tiende a comportarse como una búsqueda

local. Una particularidad del recocido simulado es que es un algoritmo sin memoria, por lo que no reúne información de los vecindarios visitados [31].

Capítulo 3

Problemas de Estudio

En la sección anterior se presentó el sobreajuste como una problemática común en los sistemas de aprendizaje automático y estadístico, en donde el modelo no es capaz de diferenciar el ruido presente en las observaciones y éste ajusta sus parámetros de tal forma que identifica el ruido como parte de las observaciones, atribuyéndolo a la no-linealidad presente en el conjunto de entrenamiento y aprendiéndolo como consecuencia. Un modelo con sobreajuste intenta aproximar una función que pase exactamente sobre las observaciones individuales en vez de buscar un patrón colectivo. Como consecuencia, cuando el modelo se entrena de nuevo con un conjunto de observaciones diferente se espera que los parámetros del modelo cambien en proporción al nivel de ruido, donde nuevamente el modelo considera el ruido como una fuente de no-linealidad diferente a la anterior. El sobreajuste conduce a una baja capacidad de pronóstico en la predicción de observaciones fuera del conjunto de entrenamiento.

En esta tesis se consideran tres problemas de estudio para analizar el efecto que tienen el ruido y la no-linealidad presentes en el conjunto de observaciones al aplicar principalmente modelos de inferencia Bayesiana (los cuales serán introducidos en el Capítulo 4), especialmente cuando el tamaño del conjunto de observaciones disminuye hasta niveles críticos. El interés en analizar este efecto recae en que la estadística Bayesiana es preferida por encima de otras metodologías clásicas por incluir de manera automática los elementos necesarios para regular la complejidad del modelo, lo que reduce el riesgo de padecer sobreajuste. De aquí que una comparación con un

método que previene el sobreajuste y que ha ganado aceptación en los últimos años, como lo es el *bootstrap*, resulte importante. Adicionalmente, la estadística Bayesiana goza de una mayor utilidad cuando el tamaño del conjunto de entrenamiento es pequeño y no es conveniente disminuir aún más su tamaño destinando un subconjunto de observaciones para validar el modelo y regular su complejidad.

Los problemas que se estudian en esta tesis provienen de diferentes áreas del conocimiento, y en dos casos corresponden con problemas de interés actual. En la Sección 3.1 se introduce el XOR continuo como nuestra primera prueba para la estadística Bayesiana. Posteriormente, en la Sección 3.2 se presenta un problema retador perteneciente al área de la bioquímica y la biología molecular, como lo es la estimación de la afinidad en acoplamientos enzimáticos. Finalmente, en la Sección 3.3 se introduce un problema concerniente a la contaminación ambiental, que involucra la depositación de metales pesados en el suelo para una región de Jura, Suiza. Los detalles específicos acerca de los conjuntos de entrenamiento de estos problemas se dejan para la Sección 5.

3.1 XOR continuo

El primer problema de estudio en esta tesis se deriva del problema de la compuerta lógica en el área de la electrónica, el cual es un dispositivo que representa la expresión física de un operador booleano en la lógica de la conmutación. Básicamente, se trata de una compuerta lógica digital que retorna como salida un valor de 0 ó 1 (i.e., *falso* o *verdadero*, respectivamente) en base a reglas lógicas sencillas dado un par de atributos booleanos. La compuerta lógica regresa como resultado *verdadero* si y sólo si uno de los atributos de entrada es *verdadero*. En cambio, si ambos atributos son *falsos* o ambos son *verdaderos*, el resultado regresado es *falso*. En resumen, la compuerta lógica regresa un valor *verdadero* si alguna de las dos entradas es *verdadera*, pero no ambas.

En la versión continua de este problema el resultado de la compuerta lógica

continúa siendo booleano, pero su respuesta es escalada (por cuestiones numéricas) a -0.5 y 0.5 para representar *falso* y *verdadero*, respectivamente. Por su parte, los atributos de entrada son ahora continuos y toman valores en el intervalo $[0.1, 1]$. Así, el problema XOR continuo consiste en la interacción de dos atributos de entrada, digamos x_1 y x_2, y el resultado que proporciona la compuerta lógica. Si x_1 y x_2 tienen un valor mayor que 0.55, o si ambos tienen un valor menor que 0.55, entonces la compuerta lógica regresa un valor *falso* ($y = -0.5$). Por el contrario, si alguno es mayor a 0.5 y el otro es menor a 0.5, entonces la compuerta lógica retorna un valor *verdadero* ($y = 0.5$). En la Figura 5.2 se presenta la separación por cuadrantes que presupone el problema, en donde la función de discriminación óptima que separa tales cuadrantes corresponde con:

$$y = f\left(-c(x_1 - 0.55)(x_2 - 0.55)\right) - 0.5, \tag{3.1}$$

siendo c un valor infinito y f una función sigmoidal, tal que:

$$f(x) = \frac{1}{1 + \exp\left(-x\right)}. \tag{3.2}$$

El XOR continuo es un problema lo suficientemente sencillo para deducir la función de discriminación analíticamente, pero al mismo tiempo representa un problema complicado de estimar para una cantidad de métodos de aprendizaje computacional, de modo que tiene la complejidad necesaria para evaluar el modelo construido mediante las técnicas de solución propuestas en esta tesis. Es por esta característica que el XOR continuo es un problema de prueba muy conocido en el área del aprendizaje automático, de modo que existen muchos otros enfoques propuestos con anterioridad para resolver este problema, como son las redes neuronales artificiales con un algoritmo genético como método de aprendizaje [55], una extensión del método *relief attribute evaluation* [33] y las máquinas de soporte vectorial [57, 58], por mencionar algunas. Dado que las observaciones carecen de ruido pero son discriminadas mediante una función no-lineal, nuestro interés en este problema reside en

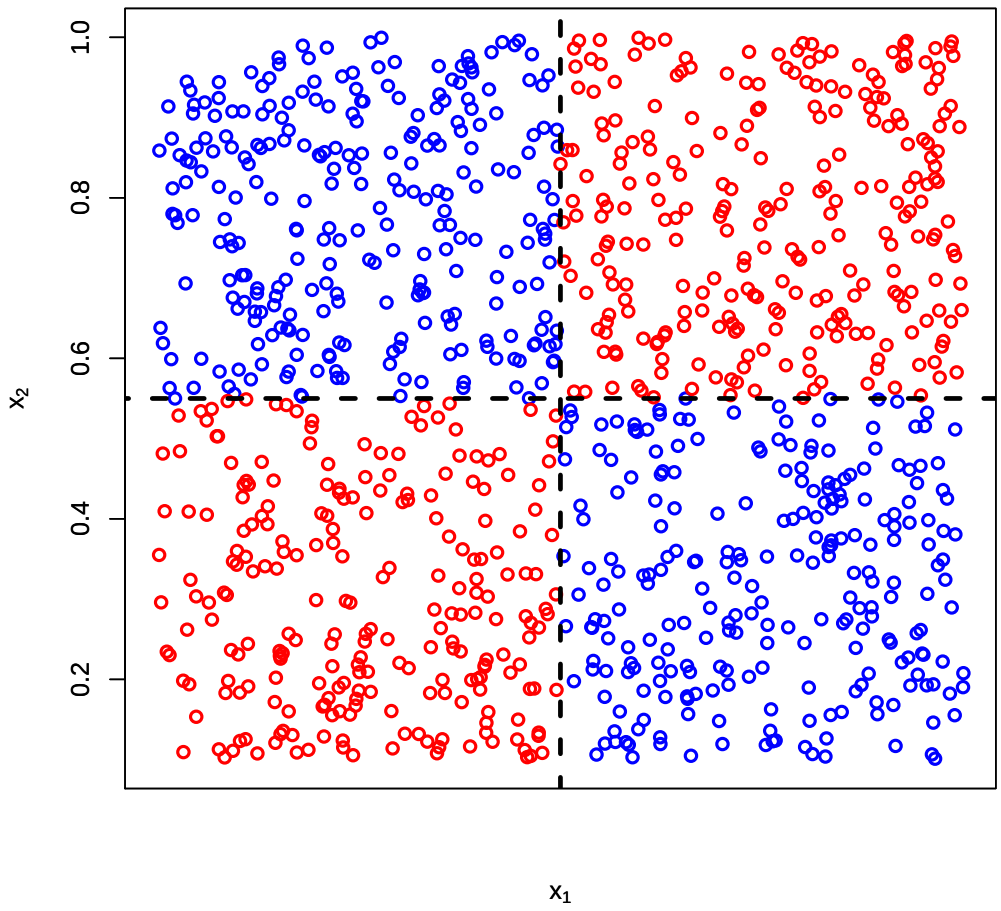


Figura 3.1: Cuadrantes en el plano formados en el problema XOR continuo. En azul se presentan los pares ordenados que proporcionan un valor *verdadero* en la compuerta lógica, mientras que en rojo se encuentran aquellos que proporcionan un valor *falso*. Las líneas punteadas indican los umbrales de separación en cuadrantes.

analizar la capacidad de pronóstico de los diferentes métodos de solución por el efecto exclusivo de la no-linealidad presente en las observaciones conforme se disminuye el tamaño del conjunto de entrenamiento.

3.2 Afinidad de acoplamientos enzimáticos

Las proteínas desempeñan un papel fundamental para la vida, siendo las biomoléculas más abundantes dentro de una célula [44]. Las funciones que éstas desempeñan a menudo implican la interacción con otras moléculas, como es el caso de su función enzimática, en donde las proteínas actúan como catalizadores para modifican la velocidad de una reacción bioquímica, recibiendo el nombre de enzimas. Un problema retador que pertenece a distintas subáreas de la biología corresponde con la estimación de la afinidad entre una enzima y un sustrato (i.e., la molécula que interacciona con la enzima) en reacciones enzimáticas. La interacción de una enzima con un sustrato es un proceso altamente selectivo, de tal forma que la unión solo puede existir cuando éstos son complementarios en forma, tamaño, carga eléctrica y caracterización química, formando un complejo tridimensional definido. Al sitio en que se da la unión se le denomina centro activo, pudiendo tener una misma enzima diferentes sitios de interacción. Es de especial interés enfatizar que la geometría del centro activo de una enzima debe corresponder con la geometría del sustrato que se une a ella, y por lo tanto, la función que desempeña una enzima se encuentra íntimamente ligada a su forma geométrica. Así, la afinidad que presenta una enzima hacia un sustrato varía de acuerdo a la geometría de ambas moléculas y tiene un impacto en su función biológica.

La afinidad es una medida de la facilidad con que una enzima y un sustrato se acoplan, por lo que es una propiedad que aparece constantemente en el área de la cinética bioquímica. Un modelo ampliamente utilizado cuando la concentración del sustrato es mayor que la concentración de la enzima es la cinética de Michaelis-Menten, la cual describe la velocidad de reacción de un gran número de reacciones

enzimáticas. La reacción en donde la enzima E se une al sustrato S para formar el complejo enzimático ES, que a su vez da lugar a la formación del producto P y a la liberación de la enzima está representada por:

$$\mathrm{E} + \mathrm{S} \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_3} \mathrm{E} + \mathrm{P}, \tag{3.3}$$

en donde k_1, k_2 y k_3 corresponden con constantes cinéticas de velocidad de reacción para las diferentes subreacciones. La velocidad de formación del producto P en el estado estable es:

$$v_P = \frac{k_3[E][S]}{K_M + [S]}, \tag{3.4}$$

donde K_M es un parámetro cinético importante, conocido como la constante de Michaelis-Menten, y es una medida relacionada estrechamente con la inversa de la afinidad de una enzima por un sustrato. A menor valor de K_M, mayor afinidad tendrá la enzima por el sustrato. Estimar adecuadamente la afinidad en un acoplamiento enzimático es un problema común para las ramas que intersectan la química, la biología y la informática.

Un método que recientemente ha ganado popularidad para obtener información sobre la estructura de una proteína consiste en el uso de arreglos proteicos, similares a los microarreglos de ADN, en donde el término *arreglo* hace énfasis en un conjunto ordenado de sitios. Cada uno de estos sitios contiene un acoplamiento enzima-sustrato diferente cuya afinidad es cuantificada mediante técnicas fluorescentes. De esta forma, se tiene un conjunto de observaciones conformado por diferentes acoplamientos enzima-sustrato y su nivel de afinidad correspondiente. Si se puede desarrollar un modelo que sea capaz de estimar adecuadamente la afinidad de una enzima bajo estudio por un sustrato, entonces el modelo puede ser útil en aplicaciones donde se necesita inhibir la función de una enzima o proteína, como lo es el diseño de fármacos, en donde se diseña una molécula que tenga una afinidad mayor que un determinado sustrato con la proteína que se desea inhibir.

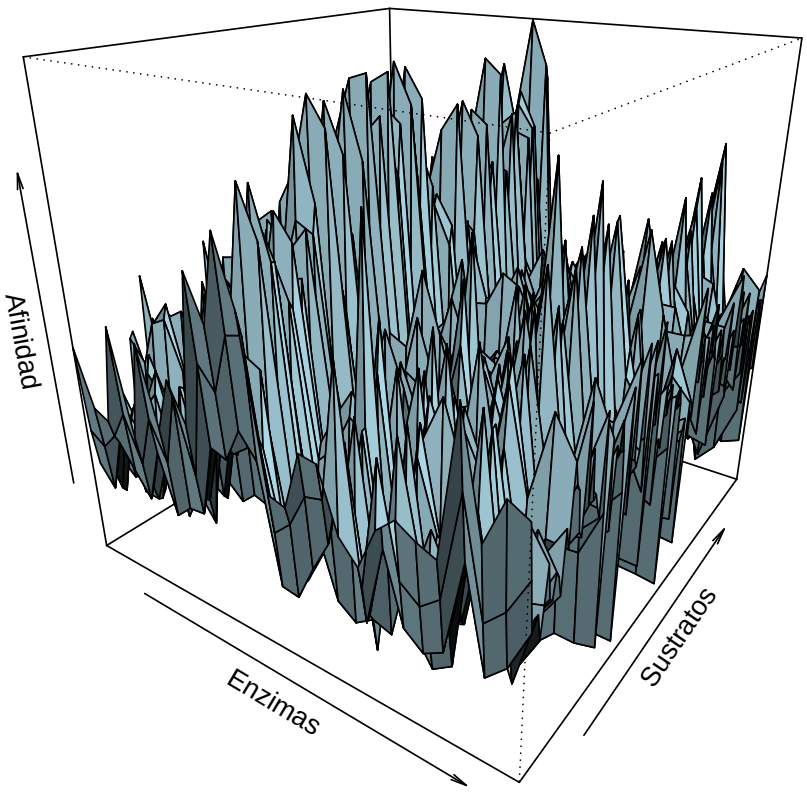
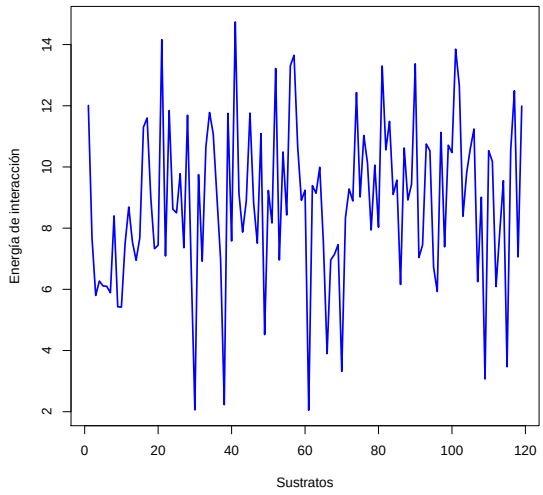
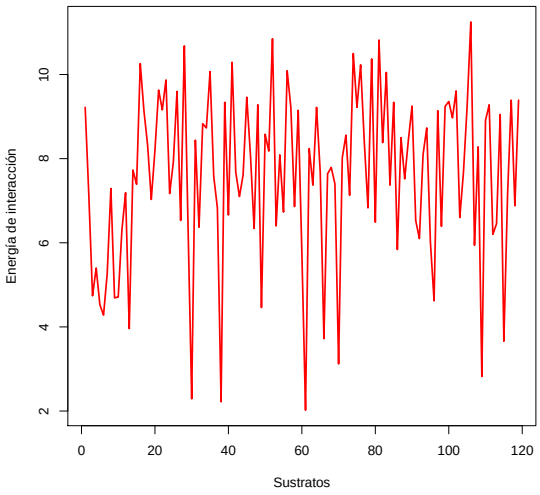


Figura 3.2: Superficie de respuesta de la afinidad para los acoplamientos enzimáticos del conjunto de observaciones utilizado en esta tesis. La altura de cada punto representa la intensidad de la afinidad (la energía de interacción) entre una enzima y un sustrato.

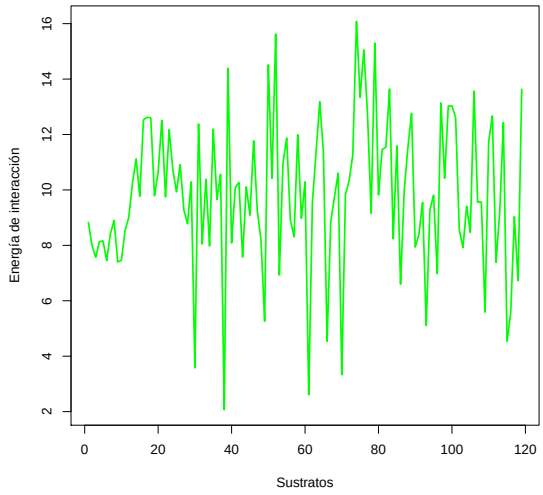
La complejidad del problema reside en que la estructura terciaria de una proteína está definida por una cantidad extraordinariamente grande de uniones débiles entre sus aminoácidos, como lo son las fuerzas electrostática, las fuerzas de Van der Waals y los puentes de hidrógeno, lo que implica que la afinidad que presenta sea difícil de estimar. En adición a esta dificultad, los sistemas bioquímicos operan a temperaturas relativamente altas, adicionando a las observaciones ruido térmico. Así, el interés en este problema consiste en analizar la capacidad de pronóstico de los métodos por el efecto combinado del ruido y de una alta no-linealidad en las observaciones conforme se disminuye el tamaño del conjunto de entrenamiento. La Figura 3.2 presenta la superficie de respuesta para el conjunto de observaciones empleado en esta tesis, el cual contiene la energía de interacción de enzimas y sustratos seleccionados de entre los metabolitos de la bacteria *Escherichia coli* como indicador de la afinidad [36]. En la Figura 3.3 se muestra la energía de interacción para diferentes proteínas al formar complejos con los diferentes sustratos del conjunto de observaciones. Ambas figuras presentan evidencia de los niveles de ruido y no-linealidad presentes en la función que se espera genere el conjunto de observaciones. Otra manera de tratar este problema es ver el arreglo como una matriz de observaciones con observaciones perdidas (que varían en cantidad al disminuir el tamaño del conjunto de entrenamiento). Distintos enfoques han sido encontrados en la literatura para resolver este tipo de problemas, como son la regresión lineal, la regresión no-lineal Bayesiana [71], la descomposición en valores singulares [34], el análisis probabilístico de componentes principales [61] y algunos modelos Bayesianos no-paramétricos [45]. De manera específica, en Nguyen et al. [46] se propone un modelo lineal con un ajuste Bayesiano de sus parámetros, para posteriormente estudiar los límites de predicción de su modelo bajo la influencia del ruido y la no-linealidad, validando su método utilizando el mismo conjunto de entrenamiento que se aplica en esta tesis. En Troyanskaya et al. [62] se presenta un estudio comparativo de diversos métodos para estimar valores perdidos en observaciones provenientes de microarreglos.



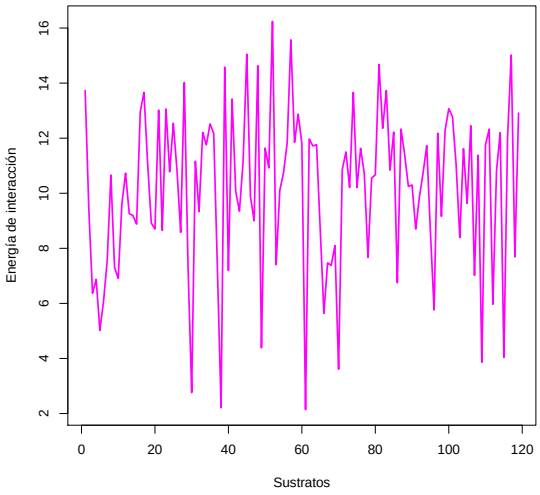
(a) Enzima E_1.



(b) Enzima E_2.



(c) Enzima E_3.



(d) Enzima E_4.

Figura 3.3: Energía de interacción para las proteínas (a) E_1, (b) E_2, (c) E_3 y (d) E_4 al unirse a los diferentes sustratos.

3.3 Concentración de metales pesados en la capa superficial del suelo

En la naturaleza, las propiedades físicas, químicas y biológicas muestran regularmente una importante heterogeneidad espacial. A pesar de esto, es posible encontrar patrones de distribución en diferentes procesos naturales. En la estadística básica, las herramientas surgen de operadores en el espacio de la variable aleatoria, mientras que en las series temporales y en el ajuste por mínimos cuadrados los operadores se trasladan al espacio de otras variables, como son el tiempo y el espacio, bajo la suposición de que las observaciones son estacionarias. Desafortunadamente, en los procesos naturales no siempre es posible realizar tal suposición, motivo por el cual se desarrolló la geoestadística como ciencia aplicada, la cual comprende el conjunto de métodos, herramientas y procedimientos que se utilizan para analizar y predecir los valores de una variable que se muestra distribuida en el espacio de forma continua, como es el caso del muestreo ambiental. Una de las razones por las que se hacen muestreos ambientales es para delimitar las zonas en donde existe contaminación por materiales potencialmente tóxicos en el suelo. Un problema de contaminación recurrente en la actualidad corresponde con la depositación de metales pesados, los cuales pueden ser desechados por la industria manufacturera, por el tráfico vehicular o incluso pueden derivarse de las piedras nativas de la región. El primer paso para determinar tales zonas es tomar muestras provenientes del suelo en diferentes ubicaciones y determinar la concentración de los metales pesados. Posteriormente, estas concentraciones son interpoladas utilizando una variedad de técnicas para estimar el grado de contaminación en las ubicaciones no muestreadas.

Los métodos de solución que serán presentados en esta tesis se aplican a un conjunto de mediciones terrestres multivariadas relacionadas con la contaminación del suelo por metales pesados en una región de 14.5 kilómetros cuadrados en Jura, Suiza [22]. Las mediciones contienen la concentración de siete metales pesados en diferentes puntos de la superficie: cadmio, cobalto, cromo, níquel, plomo y zinc.

Adicionalmente, se tiene información del tipo de roca superficial y del uso que se la da a cada ubicación muestreada. El tipo de roca corresponde con cinco posibles opciones características de Suiza, mientras que los posibles usos corresponden con bosque, pradera, pastoreo y labranza. Una descripción detallada del muestreo y las técnicas de laboratorio empleadas para hacer las mediciones se describe en [3].

En muchas situaciones existen atributos de interés que son difíciles y costosos de medir, por lo que se miden una pequeña cantidad de éstos. La falta de información es compensada con una cantidad más abundante de mediciones indirectas que se sabe están correlacionadas con los atributos de interés. Esta es una problemática que ocurre comúnmente en la industria, por ejemplo, cuando para tomar una medición se necesita destruir una pieza fabricada. De igual forma, hay metales que necesitan de un procedimiento económicamente costoso para determinar su concentración, como es el caso del cobre, de modo que se prefiere estimar su concentración mediante la concentración de otros metales que sean más accesibles de medir. El problema que aquí se plantea sigue el experimento presentado en Goovaerts et al. [22], en donde se desea estimar (i) la concentración de cadmio mediante la concentración de níquel y zinc, y (ii) la concentración de cobre mediante la concentración de plomo, níquel y zinc, además del tipo de roca superficial y del uso que se la da al suelo. Nuestro interés en este problema se encuentra en que las concentraciones de los diferentes metales están correlacionadas, de tal forma que se desea aprender de un sistema con salidas correlacionadas, lo cual representa una desventaja para los métodos que realizan las tareas de forma secuencial al compararlos con aquellos que son capaces de compartir información entre tareas [2]. Debido a esta característica, en la literatura se encuentran una gran cantidad de técnicas aplicadas a este problema, como son los procesos Gaussianos [2], los procesos Gaussianos con multi-kernels [40], las redes de regresión de procesos Gaussianos [66], los procesos de convolución [2] y enfoques no-paramétricos de covarianza cruzada [70].

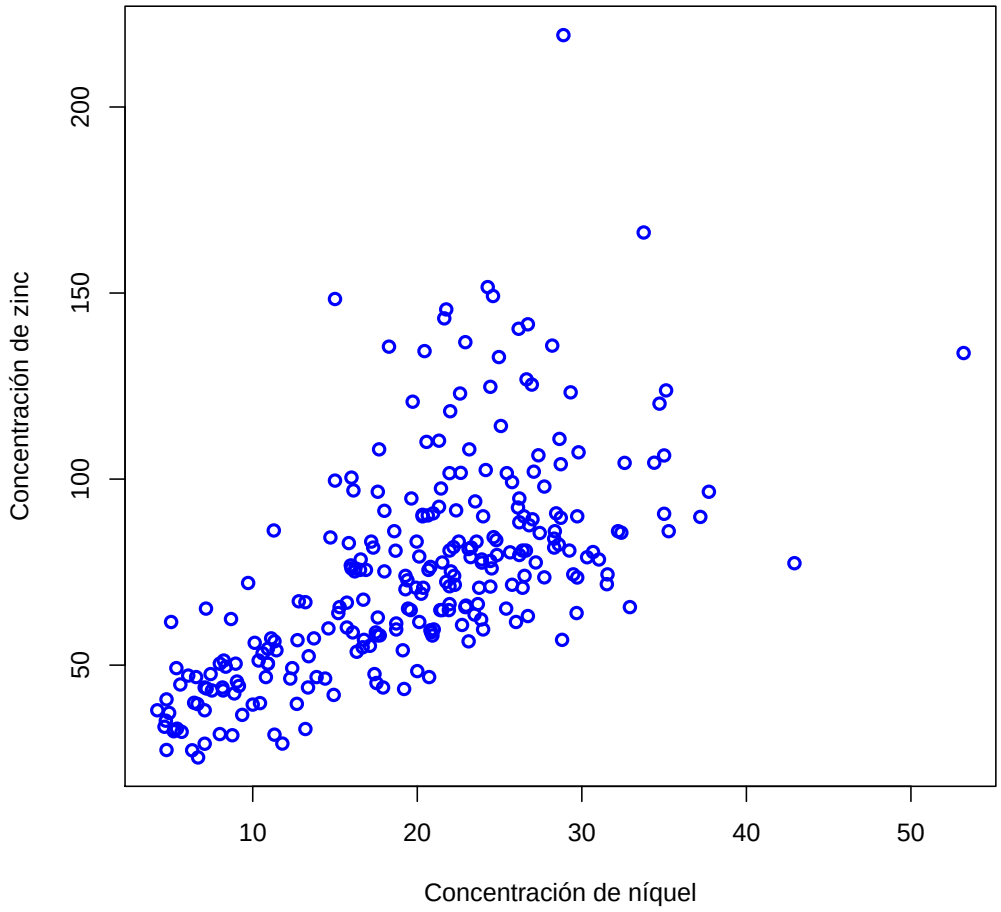


Figura 3.4: Efecto de la correlación existente entre las concentraciones de níquel y zinc para el conjunto de observaciones bajo estudio.

Capítulo 4

Métodos de Solución

El objetivo principal de esta tesis es aplicar métodos Bayesianos estadísticos y de aprendizaje automático en problemas retadores de estudio como lo son el XOR continuo, la afinidad en acoplamientos enzimáticos y la concentración de metales pesados en la capa superficial del suelo, conforme se disminuye el tamaño del conjunto de observaciones de entrenamiento, de modo que sea posible analizar la pérdida de precisión de tales métodos como efecto que tiene el tamaño muestral, así como el ruido y la no-linealidad presentes en las observaciones. De igual forma, se desea analizar el desempeño de las técnicas Bayesianas en comparación con una técnica que ha ganado aceptación y popularidad en los últimos años por ser capaz de resolver problemas en donde otros métodos han fallado: el *bootstrap*, una herramienta estadística que produce conjuntos de observaciones adicionales mediante la muestra original para estimar la distribución muestral de un estadístico. En este capítulo se presentan los métodos de solución. En la Sección 4.1 se introducen las redes neuronales artificiales como modelos paramétricos de aproximación, cuya modelación paramétrica sirve de motivación para introducir los procesos Gaussianos en la Sección 4.2. Posteriormente, en la Sección 4.3 se presentan las mezclas infinitas de Gaussianas, para finalizar con una descripción del *bootstrap* en la Sección 4.4. En el Apéndice A se presenta una descripción acerca de la implementación computacional de los métodos Bayesianos aquí descritos. Esta referencia es particularmente útil para ilustrar el método de mezclas infinitas de Gaussianas, ya que no hay ningún paquete que cuente con esta implementación hasta donde nosotros sabemos.

4.1 Redes Neuronales Artificiales

En años recientes se ha visto un gran número de avances en el desarrollo de los sistemas inteligentes, algunos inspirados por las redes neuronales biológicas. De entre los aportes más estudiados se encuentran las redes neuronales artificiales. Investigadores de diferentes áreas del conocimiento diseñan redes neuronales artificiales para resolver una amplia gama de problemas que abarcan disciplinas como el reconocimiento de patrones, la psicología cognitiva y el control automático, así como labores de predicción, de optimización y de memoria asociativa [51]. A pesar de que se han propuesto soluciones convencionales que funcionan adecuadamente para situaciones restringidas de estos problemas, ninguna de ellas es lo suficientemente flexible para desempeñar su labor fuera del dominio para el que fueron diseñadas [25]. Las redes neuronales artificiales son una alternativa flexible e inteligente que pueden beneficiar un gran número de aplicaciones, debido principalmente a que poseen la habilidad de aprender ciertas propiedades de un conjunto de observaciones, por lo que parecen funcionar extremadamente bien incluso para problemas en donde los métodos estadísticos convencionales fallan [25].

Las redes neuronales artificiales, o simplemente redes neuronales para los fines de esta tesis, fueron originalmente motivadas por un interés en imitar algunos de los métodos de procesamiento encontrados en el cerebro orgánico. El cerebro es un computador paralelo altamente complejo y no-lineal que tiene la habilidad de organizar sus constituyentes estructurales, conocidos como neuronas, de tal forma que son capaces de realizar ciertos cálculos de forma más rápida que la computadora más veloz construida hasta el momento. A la unión intercelular entre dos neuronas se le conoce como sinapsis, y es el medio por el cual se lleva a cabo la transmisión de información, mediante impulsos nerviosos. Así como el cerebro es capaz de efectuar tareas de diversa complejidad, las redes neuronales han demostrado ser capaces de resolver problemas complejos de diferente índole [51]. Antes de continuar con las propiedades de las redes neuronales y de como éstas son capaces de aprender de un

conjunto de observaciones, es conveniente definir lo que es una red neuronal.

Una red neuronal es un conjunto de neuronas artificiales interconectadas entre sí, las cuales funcionan como procesadores independientes que regresan una función acotada de su salida total [54]. Tales neuronas están acomodadas en capas, de las cuales se distinguen tres tipos: la capa de entrada, la capa de salida y una o varias capas ocultas. El resto de esta sección se dedica a introducir la arquitectura de las redes neuronales (Sección 4.1.1), la forma en que éstas computan su salida (Sección 4.1.2), la manera en que el aprendizaje se lleva a cabo (Sección 4.1.3) y el muestreo aleatorio que da como resultado el valor esperado a la salida de la red para un conjunto de observaciones (Sección 4.1.4).

4.1.1 Arquitectura de una red neuronal artificial

La aplicación de las redes neuronales en problemas de aproximación de funciones e inferencia es un enfoque paramétrico [37]. Esto implica que se hace la suposición de que existe una función no-lineal, digamos $y_k(\boldsymbol{x})$, que está detrás del conjunto de observaciones y objetivos $\{\boldsymbol{x}_n, t_n^k\}_{n=1}^N$, donde $k \in \{1, \ldots, u\}$ y corresponde con la dimensionalidad de los objetivos en $\boldsymbol{t}_n$. En esta nomenclatura $\boldsymbol{x}_n$ representa el vector formado por el conjunto de atributos de la observación n, de tal modo que $\boldsymbol{x}_n = (x_n^1, \ldots, x_n^p)^T$. De manera semejante, $\boldsymbol{t}_n$ corresponde con el vector formado por el conjunto de objetivos de la misma observación n, de tal forma que $\boldsymbol{t}_n = (t_n^1, \ldots, t_n^u)^T$. Cada función desconocida $y_k(\boldsymbol{x})$ es parametrizada por medio de los parámetros $\boldsymbol{w}_k$, conocidos como pesos en la literatura de las redes neuronales, y los cuales están íntimamente relacionadas con la intensidad de la conexión sináptica de una neurona biológica. La capacidad de aprendizaje de una red neuronal consiste en aproximar $y_k(\boldsymbol{x})$ por medio de la función parametrizada $y_k(\boldsymbol{x}; \boldsymbol{w}_k)$.

En la Figura 4.1 se muestra la estructura de una red neuronal tipo prealimentación con una única capa oculta. El término prealimentación implica un tipo de red neuronal en que las conexiones sinápticas están dirigidas de las entradas a las salidas

de la red, es decir, la información fluye siempre de la capa de entrada a la capa de salida. Al tamaño de la red neuronal se le denomina arquitectura, y comprende la cantidad de capas ocultas que contiene, además de la cantidad de neuronas en cada una de estas capas. Las redes neuronales monocapa (i.e., con una única capa oculta) son las más comúnmente encontradas en literatura [37], y son las que se utilizan en esta tesis como método de solución. La arquitectura de una red neuronal varía de acuerdo al conjunto de observaciones del que se quiere aprender. El tamaño de la capa de entrada depende completamente de la dimensionalidad, p, de las observaciones (o atributos), $\boldsymbol{x}_n$. De igual manera, el tamaño de la capa de salida depende exclusivamente de la dimensionalidad, u, de los objetivos, $\boldsymbol{t}_n$. Cada neurona en la capa de salida tiene asociado un peso del conjunto $\{w_y^k\}_{k=1}^u$ (en rojo en la Figura 4.1). La capa oculta, por su parte, es interna a la red y no tiene contacto directo con el exterior. El tamaño de esta capa no está en función de las observaciones ni restringido de forma alguna, sino que por el contrario, conforme el tamaño de esta capa aumenta (o conforme se adicionan capas ocultas a la arquitectura) se confiere habilidad a la red neuronal para extraer propiedades estadísticas de mayor orden del sistema bajo estudio. Asumiendo una arquitectura con una única capa oculta y q unidades de procesamiento, cada neurona en esta capa tiene asociado un peso del conjunto $\{w_h^j\}_{j=1}^q$ (en verde en la Figura 4.1). Además, cada conexión sináptica entre las neuronas de la capa de entrada y las neuronas de la capa oculta tiene asociada un peso del conjunto $\{w_{eh}^{ij}\}$, en donde $ij = \{1, \ldots, p \cdot q\}$ (en azul en la Figura 4.1). De manera semejante, cada conexión sináptica entre las neuronas de la capa oculta y las neuronas de la capa de salida tiene asociada un peso del conjunto $\{w_{hy}^{jk}\}$, en donde $jk = \{1, \ldots, q \cdot u\}$ (en magenta en la Figura 4.1).

A pesar de que al aumentar el tamaño de la capa oculta la red neuronal adquiere habilidad para extraer propiedades estadísticas de mayor orden, Neal [42] ha demostrado que existe un límite en el cual las propiedades estadísticas de las funciones generadas al aleatorizar los pesos de una red neuronal son independientes del número de unidades ocultas, de modo que la complejidad de las funciones parametrizadas se vuelve independiente del número de parámetros en el modelo.

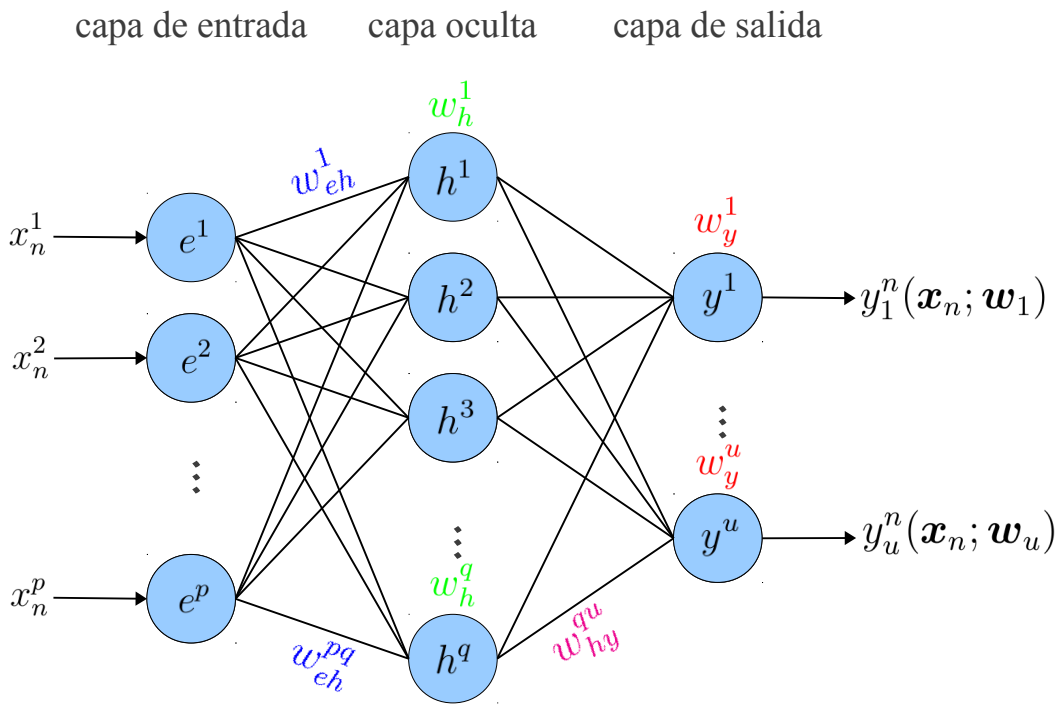


Figura 4.1: Estructura de una red neuronal tipo prealimentación con una única capa oculta. En verde y rojo se muestran los pesos relacionados con la capa oculta y la capa de salida, respectivamente, mientras que en azul y magenta se presentan los pesos asociados a las conexiones sinápticas entre las capas entrada-oculta y oculta-salida, respectivamente.

Consecuencias más profundas de estas observaciones serán analizadas en la Sección 4.2, cuando se introduzcan los procesos Gaussianos como métodos de solución. Por lo pronto, basta comprender que al aumentar la cantidad de neuronas en la capa oculta por encima de un cierto límite, el cual está en función tanto de la complejidad del problema como de la cantidad y calidad de las observaciones disponibles, la capacidad de inferencia de la red neuronal se mantiene aproximadamente constante.

4.1.2 Salida emitida por una red neuronal artificial

La forma en que la información es recibida, procesada y transmitida entre las diferentes capas de neuronas para evaluar la salida de la red neuronal, suponiendo que se conocen los pesos $\boldsymbol{w}$, se detalla a continuación. Dada la observación n, las neuronas de entrada se encargan de recibir el valor numérico de los p atributos en $\boldsymbol{x}_n$ para posteriormente transmitir esta información inalterada a las neuronas en la capa oculta utilizando las conexiones sinápticas que las unen. La información que la neurona e^i en la capa de entrada recibe y que posteriormente transmite a las neuronas en la capa oculta corresponde con la observación asociada a ella:

$$e^i_{\text{transmisión}} = e^i_{\text{recepción}} = x^i_n \qquad \forall i \in \{1, \ldots, p\}. \tag{4.1}$$

Por su parte, cada neurona en la capa oculta recibe información de todas las neuronas en la capa anterior por medio de sus conexiones sinápticas. Dado que cada conexión sináptica tiene un peso asociado, w^{ij}_{eh}, la información total que recibe la neurona h^j en la capa oculta es una suma ponderada de la información que recibe por medio de sus conexiones sinápticas. Esto es,

$$h^j_{\text{recepción}} = \sum_{i=1}^{p} w^{ij}_{eh} \cdot e^i_{\text{transmisión}} = \sum_{i=1}^{p} w^{ij}_{eh} \cdot x^i_n \qquad \forall j \in \{1, \ldots, q\}. \tag{4.2}$$

Ahora es necesario introducir una nueva propiedad de los sistemas artificiales y su analogía con el sistema biológico. Una neurona biológica puede estar activa o inacti-

va, es decir, puede estar o no excitada, lo que implica que tiene asociado un estado de activación. Las neuronas artificiales también tienen estados de activación asociados. Algunas tienen solamente un estado de activación binario, como las neuronas biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado [25]. Se denomina función de activación [69], $\varphi(\cdot)$, a la función encargada de evaluar el estado de actividad de una neurona artificial, transformando la entrada de tal neurona en un estado de activación. Para esta tesis se utiliza la tangente hiperbólica como función de activación (ec. 4.3), pues además de ser una de las funciones de activación más conocidas y estudiadas, el proceso de convergencia se realiza más rápidamente en comparación con otras funciones de activación [69]. Como resultado, los estados de activación de una neurona están comprendidos dentro del intervalo $[-1, 1]$, tal y como se muestra en la Figura 4.2. El valor de g en (4.3) indica la pendiente de la función de activación, como también puede observarse en la Figura 4.2.

$$\varphi(x) = \tanh(g \cdot x). \tag{4.3}$$

Además de la función de activación, al modelo de una red neuronal se suelen añadir algunas neuronas más, denominadas *bias*. Este tipo de neuronas siempre emiten el valor unitario y están conectadas con todas las neuronas de la capa siguiente. La función de estas neuronas en las redes neuronales es importante, puesto que generalmente no se conocen los aspectos internos del sistema del que se quiere aprender. Sin importar la arquitectura de la red neuronal, algunas funciones no pueden aprenderse sin el uso de neuronas tipo *bias*. Considérese, por ejemplo, el caso en que todas las variables de entrada tienen un valor de cero. Sin la existencia de las neuronas tipo *bias*, la única salida posible de la red neuronal es cero, ya que $\varphi(0) = 0$. Sin embargo, añadiendo el peso w de una neurona tipo *bias*, la salida de la red neuronal puede ser fácilmente escalada al valor objetivo, ya que $\varphi(0 + w) \neq 0$. Así, a la información que recibe la neurona k en la capa oculta se le adiciona un peso, w_h^j (en verde en la Figura 4.1), que simula el comportamiento de estas neuronas tipo *bias*. Posteriormente se aplica la función de activación para determinar el estado de activación de

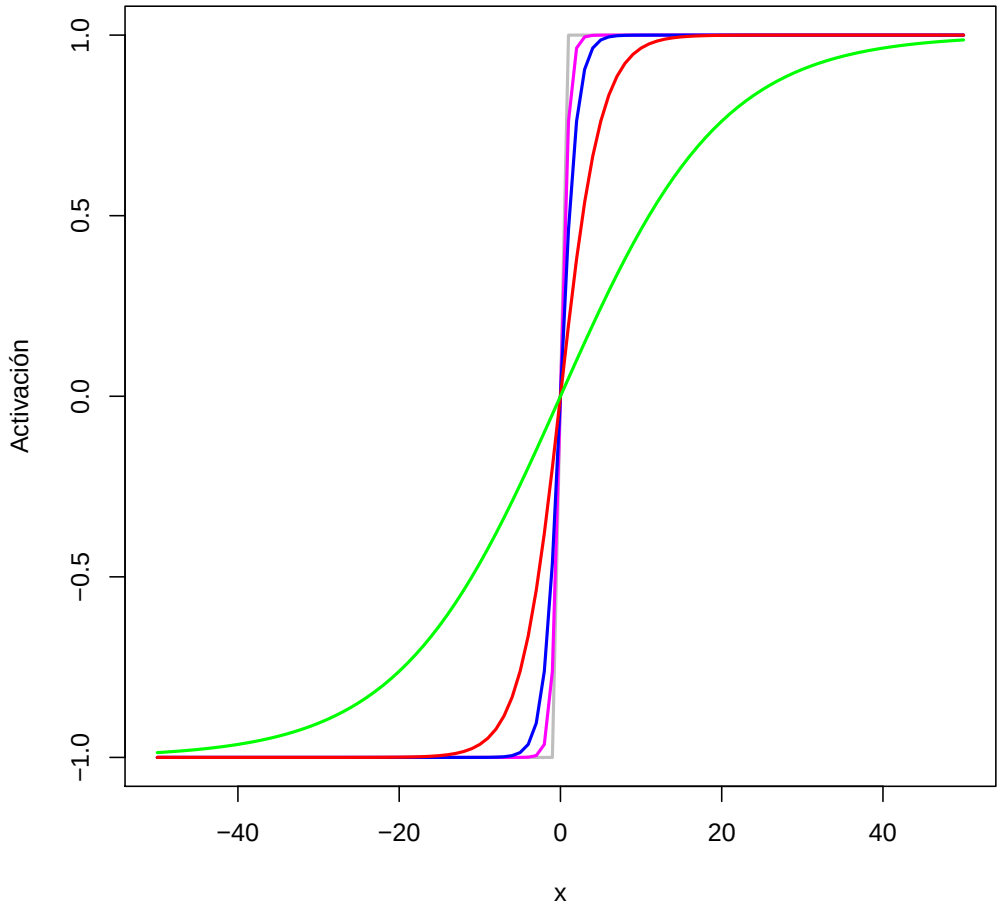


Figura 4.2: Función de activación tangente hiperbólica. En verde el caso en que $g = 0.05$, en rojo $g = 0.2$, en azul $g = 0.5$, en magenta $g = 1.0$ y en gris $g = 5.0$.

la neurona. La información que la neurona h^j transmite a la capa de salida es:

$$h^j_{\text{transmisión}} = \tanh\left[\sum_{i=1}^{p} w^{ij}_{eh} \cdot x^i_n + w^j_h\right] \qquad \forall j. \tag{4.4}$$

De manera semejante, cada neurona en la capa de salida recibe información de todas las neuronas en la capa anterior por medio de sus conexiones sinápticas. Como cada conexión sináptica tiene asociada un peso, w^{jk}_{hy}, la información que recibe la neurona y^k en la capa de salida es una suma ponderada de la información que recibe por medio de sus conexiones sinápticas. Es decir,

$$y^k_{\text{recepción}} = \sum_{j=1}^{q} w^{jk}_{hy} \cdot h^j_{\text{transmisión}} \qquad \forall k \in \{1, \ldots, u\}. \tag{4.5}$$

Finalmente, cada una de las neuronas en la capa de salida computan su valor de salida. A la información ponderada que la neurona y^k recibe por medio de sus conexiones sinápticas se le adiciona el peso w^y_k (en rojo en la Figura 4.1), equivalente a la neurona tipo *bias*:

$$y^k_{\text{transmisión}} = \sum_{j=1}^{q} w^{jk}_{hy} \cdot h^j_{\text{transmisión}} + w^k_y \qquad \forall k. \tag{4.6}$$

La información que transmiten las neuronas en la capa de salida corresponde con la salida de la red neuronal, de tal forma que (4.6) es equivalente al resultado de la función parametrizada $y^n_k(\boldsymbol{x}_n; \boldsymbol{w}_k)$, en donde $\boldsymbol{w}_k$ indica el vector formado por los pesos que son parte del modelo para la neurona de salida y^k, es decir, todos los pesos con excepción de aquellos en la capa de salida que son diferentes a k. Tal conjunto corresponde con:

$$\boldsymbol{w}_k = (w^1_{eh}, \ldots, w^{pq}_{eh}, w^1_h, \ldots, w^q_h, w^1_{hy}, \ldots, w^{qu}_{hy}, w^k_y)^T \tag{4.7}$$

Agrupando las ecuaciones de la red neuronal desarrolladas hasta el momento para la observación n, la parametrización que se hace sobre la función $y^n_k(\boldsymbol{x}_n)$ es:

$$y_k^n(\boldsymbol{x}_n;\boldsymbol{w}_k) = \sum_{j=1}^{q} w_{hy}^{jk} \cdot \tanh\left[\sum_{i=1}^{p} w_{eh}^{ij} \cdot x_n^i + w_h^j\right] + w_y^k \qquad \forall k, n. \tag{4.8}$$

De esta forma, $y_k^n(\boldsymbol{x}_n;\boldsymbol{w}_k)$ en (4.8) corresponde con la aproximación paramétrica del objetivo t_n^k. Es importante mencionar que la aplicación de las redes neuronales artificiales como métodos paramétricos de aproximación de funciones se encuentra respaldada por el teorema de aproximación universal (Teorema 4.1).

Teorema 4.1 (Teorema de aproximación universal) *Las redes neuronales artificiales tipo prealimentación con una función de activación arbitraria, una única capa oculta y un número finito de neuronas en tal capa fungen como aproximadores universales para un subconjunto compacto en* $\mathbb{R}^n$.

4.1.3 Aprendizaje de una red neuronal artificial

Una vez que se ha decidido la arquitectura de la red neuronal se procede a inferir las funciones parametrizadas $y_k(\boldsymbol{x};\boldsymbol{w}_k)$ mediante la inferencia de los pesos, $\boldsymbol{w}_k$ [42]. Hasta ahora se ha hablado acerca de cómo evaluar la salida de una red neuronal cuando se conocen sus pesos, los cuales se adaptan al conjunto de observaciones mediante una función de aprendizaje [25]. De esta manera, la función de aprendizaje define la forma en que los pesos de una red neuronal varían con respecto al tiempo. En el caso de una red neuronal para aprendizaje supervisado (i.e., cuando el conjunto de observaciones tiene variables de salida definidas) la función de aprendizaje debe incluir una función de costo que cuantifique la diferencia de las salidas de la red neuronal con respecto a los objetivos del conjunto de observaciones [37].

Desde un punto de vista frecuentista, el método de entrenamiento más conocido para determinar el conjunto de pesos óptimo es el algoritmo de retropropagación [26]. Por otro lado, desde un punto de vista Bayesiano, lo que se desea es estimar la distribución probabilística de los pesos dadas las observaciones [42]. Si se asume que cada elemento en $\boldsymbol{t}^k = (t_1^k, \ldots, t_N^k)^T$ difiere del elemento correspondiente en

$\boldsymbol{y}_k(\boldsymbol{X}_N;\boldsymbol{w}_k) = [y_k^1(\boldsymbol{x}_1;\boldsymbol{w}_k),\ldots,y_k^N(\boldsymbol{x}_N;\boldsymbol{w}_k)]^T$ por un ruido aditivo de varianza σ_v^2,

$$\boldsymbol{t}^k = \boldsymbol{y}_k(\boldsymbol{X}_N;\boldsymbol{w}_k) + \mathcal{N}(\boldsymbol{0},\sigma_v^2\boldsymbol{I}_N) \qquad \forall k, \tag{4.9}$$

donde $\boldsymbol{I}_N$ representa la matriz identidad de tamaño $(N\times N)$, entonces la distribución condicional de las observaciones dados los pesos sigue también una distribución Gaussiana. Sean $\boldsymbol{X}_N$ el conjunto formado por las N observaciones, tal que $\boldsymbol{X}_N = \{\boldsymbol{x}_n\}_{n=1}^N$ y $\boldsymbol{T}_N$ el conjunto formado por los N objetivos, de modo que $\boldsymbol{T}_N = \{\boldsymbol{t}_n\}_{n=1}^N$, entonces la distribución de probabilidad de los objetivos dados los pesos y las observaciones viene dada por:

$$p(\boldsymbol{T}_N|\boldsymbol{w},\boldsymbol{X}_N) = \prod_{k=1}^{u}\mathcal{N}\left(\boldsymbol{y}_k(\boldsymbol{X}_N;\boldsymbol{w}_k),\sigma_v^2 I_N\right) \tag{4.10}$$

$$= \frac{1}{(2\pi\sigma_v^2)^{Nu/2}}\exp\left[-\frac{1}{2\sigma_v^2}\sum_{n=1}^{N}\sum_{k=1}^{u}\left[t_n^k - y_k^n(\boldsymbol{x}_n;\boldsymbol{w}_k)\right]^2\right]. \tag{4.11}$$

Este término representa la probabilidad de obtener los objetivos en $\boldsymbol{T}_N$ dados los pesos $\boldsymbol{w}$. A su vez, la distribución a priori seleccionada para los pesos en esta tesis es uniforme con dominio $[-1,1]$. Es decir,

$$p(\boldsymbol{w}) \sim \mathcal{U}[-1,1]. \tag{4.12}$$

De acuerdo al teorema de Bayes (Teorema 2.1), la distribución a priori de los pesos y su verosimilitud (i.e., la distribución de los objetivos dados los pesos y las observaciones) se combinan para estimar la distribución posterior de los pesos dadas las observaciones. Es decir,

$$p(\boldsymbol{w}|\boldsymbol{X}_N,\boldsymbol{T}_N) = \frac{p(\boldsymbol{T}_N|\boldsymbol{w},\boldsymbol{X}_N)p(\boldsymbol{w})}{p(\boldsymbol{T}_N|\boldsymbol{X}_N)}. \tag{4.13}$$

Si la dependencia de $\boldsymbol{y}(\boldsymbol{x};\boldsymbol{w}) = (y_1(\boldsymbol{x};\boldsymbol{w}_1),\ldots,y_u(\boldsymbol{x};\boldsymbol{w}_u))^T$ en $\boldsymbol{w}$ es no-lineal, entonces generalmente la distribución posterior $p(\boldsymbol{w}|\boldsymbol{X}_N,\boldsymbol{T}_N)$ no sigue una distribución

Gaussiana. Dado que tanto la densidad de probabilidad de una distribución uniforme como la constante de normalización en (4.13) permanecen constantes en todo el dominio de los pesos, la distribución posterior de los pesos dadas las observaciones es proporcional a la función de verosimilitud desarrollada en (4.11):

$$p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N) \propto p(\boldsymbol{T}_N|\boldsymbol{w}, \boldsymbol{X}_N). \tag{4.14}$$

Adicionalmente, se suele adoptar la función logarítmica de la distribución posterior como función de aprendizaje, debido a cuestiones numéricas relacionadas con el muestreo aleatorio. La función logarítmica de (4.14) es:

$$\ln p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N) \propto -\frac{1}{2\sigma_v^2} \sum_{n=1}^{N} \sum_{k=1}^{u} \left[t_n^k - y_k^n(\boldsymbol{x}_n; \boldsymbol{w}_k)\right]^2 - \frac{N}{2} \ln(2\pi\sigma_v^2). \tag{4.15}$$

Esta función logarítmica representa la función de aprendizaje de nuestra implementación de una red neuronal propuesta como método de solución. A esta función se le conoce en la literatura como función *logposterior* [42].

4.1.4 Muestreo de los pesos de una red neuronal

En inferencia Bayesiana para una red neuronal, los dos puntos clave para realizar inferencia son el desarrollo de un modelo probabilista para los pesos y el muestreo de esta distribución de probabilidad para realizar inferencias dadas las observaciones disponibles [37]. Habiendo definido el modelo probabilista, el cual consiste en la distribución de los pesos dadas las observaciones en (4.14), las predicciones se realizan haciendo una marginalización de éste con respecto a los pesos. Sea $\boldsymbol{x}_f$ una observación para la cual se desea inferir los objetivos $\boldsymbol{t}_f$, tal marginalización corresponde con:

$$p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{X}_N, \boldsymbol{T}_N) = \int p(\boldsymbol{t}_f|\boldsymbol{w}, \boldsymbol{x}_f) p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N) d\boldsymbol{w}. \tag{4.16}$$

A la observación $\boldsymbol{x}_f$ sobre la que se desea hacer inferencia se le denomina observación de prueba. En esta tesis se aplica el muestreo de Gibbs con paso Metrópolis descrito en la Sección 2.2.2 para generar muestras de la distribución posterior de los pesos y evaluar así la marginalización en (4.16) mediante integración numérica [37]. Utilizando el muestreo de Gibbs con paso Metrópolis se generan M muestras $\{\boldsymbol{w}^1, \ldots, \boldsymbol{w}^M\}$ de los pesos, con la intención de que éstas provengan de la distribución posterior $p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N)$ en (4.14), muestreando de su distribución posterior logarítmica, $\ln p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N)$, en (4.15). Utilizando las muestras generadas, la aproximación numérica a la marginalización en (4.16) está dada por:

$$p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{X}_N, \boldsymbol{T}_N) \simeq \frac{1}{M}\sum_{m=1}^{M} p(\boldsymbol{t}_f|\boldsymbol{w}^m, \boldsymbol{x}_f). \tag{4.17}$$

Mediante esta aproximación se evalúa numéricamente la integral relativa a la marginalización. Un enfoque semejante es utilizar las muestras $\{\boldsymbol{w}^1, \ldots, \boldsymbol{w}^M\}$ obtenidas mediante el muestreo de Gibbs con paso Metrópolis para evaluar M salidas de la red neuronal. De esta forma, la predicción para el valor del objetivo $\boldsymbol{t}_f$ corresponde con el valor esperado a la salida de la red neuronal dada la observación $\boldsymbol{x}_f$:

$$\langle \boldsymbol{t}_f|\boldsymbol{x}_f \rangle = \frac{1}{M}\sum_{m=1}^{M} \boldsymbol{y}^f(\boldsymbol{x}_f; \boldsymbol{w}^m). \tag{4.18}$$

En la siguiente sección se presenta otra perspectiva del desarrollo de un enfoque paramétrico y se retoman los resultados obtenidos por Neal [42] al aumentar el tamaño de la capa oculta de una red neuronal, lo que sirve como motivación para introducir los procesos Gaussianos como métodos de solución.

4.2 Procesos Gaussianos

En la Sección 4.1 se describió la implementación basada en redes neuronales artificiales como uno de los métodos de solución aplicados en esta tesis. Las redes

neuronales son técnicas adaptivas de aprendizaje computacional cuya aplicación en problemas complejos de diferentes áreas del conocimiento ha derivado en resultados satisfactorios que han sido reportado en la literatura [51], por lo cual se han seleccionado como uno de los métodos de solución en esta tesis. A su vez, la aplicación de las redes neuronales artificiales para aproximar funciones y como herramientas de inferencia se encuentra respaldada por el teorema de aproximación universal (Teorema 4.1). Sin embargo, Neal [42] ha demostrado que en el límite cuando el tamaño de la red neuronal tiende a infinito, esto es, cuando el número de neuronas en la capa oculta tiende a infinito, la distribución a priori sobre funciones no-lineales implicada por las redes neuronales con aprendizaje Bayesiano cae dentro de la clase de distribuciones de probabilidad conocida como procesos Gaussianos. Las observaciones de Neal han motivado la idea de descartar el uso de las redes parametrizadas y trabajar directamente con procesos Gaussianos.

El concepto detrás de las redes neuronales consiste en suponer que existe una función no-lineal $y_k(\boldsymbol{x})$ que está detrás del conjunto de observaciones de entrenamiento $\{\boldsymbol{x}_n, t_n^k\}_{n=1}^N$. Esta función es aproximada mediante una función parametrizada por los pesos $\boldsymbol{w}_k$ de la red neuronal, $y_k(\boldsymbol{x}; \boldsymbol{w}_k)$. De acuerdo a (4.13), para inferir futuros valores de la variable de salida (i.e., futuros valores de $\boldsymbol{t}^k$) sólo importan la distribución a priori de los pesos y el modelo que se asume para el ruido en las observaciones (i.e., la verosimilitud de los pesos dadas las observaciones). La idea detrás del modelamiento con procesos Gaussianos es insertar una distribución a priori directamente en el espacio de funciones, sin parametrizar $y_k(\boldsymbol{x})$. La distribución a priori más sencilla sobre el espacio de funciones corresponde con un proceso Gaussiano. El resto de esta sección se dedica a relacionar los procesos Gaussianos con un enfoque paramétrico (Sección 4.2.1), para posteriormente introducir de manera formal los procesos Gaussianos (Sección 4.2.2), su generalización como modelo de predicción (Sección 4.2.3), la definición de su función de covarianza (Sección 4.2.4) y la forma en que el aprendizaje se lleva a cabo (Sección 4.2.5).

4.2.1 De un modelo paramétrico a un proceso Gaussiano

Como se mencionó al inicio de esta sección, al aplicar una aproximación paramétrica a un problema de inferencia se expresa la función desconocida que se desea estimar, $y_k(\boldsymbol{x})$, en términos de una función no-lineal parametrizada por los parámetros $\boldsymbol{w}_k$, $y_k(\boldsymbol{x};\boldsymbol{w}_k)$. En el caso de las redes neuronales estos parámetros corresponden con los pesos. Dado que toda función continua en el espacio de funciones puede ser representada como una combinación lineal de funciones base [37], es posible utilizar una serie de funciones base, digamos $\{\pi^h(\boldsymbol{x})\}_{h=1}^{H}$, como modelo de aproximación para $y_k(\boldsymbol{x};\boldsymbol{w}_k)$:

$$y_k(\boldsymbol{x};\boldsymbol{w}_k) = \sum_{h=1}^{H} w_k^h \pi^h(\boldsymbol{x}) \qquad \forall k. \tag{4.19}$$

Si las funciones base son funciones no-lineales de $\boldsymbol{x}$, entonces $y_k(\boldsymbol{x};\boldsymbol{w}_k)$ es una función no-lineal de $\boldsymbol{x}$. Suponiendo que se ha seleccionado la parametrización a efectuar (e.g., la topología de una red neuronal) se procede a inferir las funciones en $\boldsymbol{y}(\boldsymbol{x};\boldsymbol{w})$ mediante la inferencia de los parámetros $\boldsymbol{w}$, utilizando la distribución posterior de los parámetros dadas las observaciones, desarrollada en (4.13). Como se mencionó en la Sección 4.1.3, si la función $y_k(\boldsymbol{x};\boldsymbol{w}_k)$ depende de los parámetros $\boldsymbol{w}_k$ en forma no-lineal, entonces generalmente la distribución posterior no es una distribución Gaussiana. El enfoque empleado en esta tesis para sobrellevar tal inconveniente al aplicar redes neuronales es utilizar un método tipo Monte Carlo para generar muestras que provengan de la distribución posterior $p(\boldsymbol{w}|\boldsymbol{X}_N,\boldsymbol{T}_N)$ y evaluar numéricamente la integral presente en la marginalización de los parámetros (ec. 4.16).

Ahora bien, si se emplean H funciones base como método de aproximación para $y_k(\boldsymbol{x};\boldsymbol{w}_k)$ y éstas son evaluadas en cada una de las N observaciones del conjunto de entrenamiento $\boldsymbol{X}_N$, entonces es posible crear una matriz $\boldsymbol{R}$ de dimensión $(N\times H)$ que contenga los valores de las H funciones base al ser evaluadas en las N observaciones del conjunto de entrenamiento. Así, el elemento R_n^h de tal matriz corresponde con la

evaluación de la función base π^h en la observación $\boldsymbol{x}_n$. Es decir,

$$R_n^h = \pi^h(\boldsymbol{x}_n) \qquad \forall h, n. \tag{4.20}$$

Sea $\boldsymbol{y}_k$ un vector que represente los valores de la k-ésima función desconocida evaluada en las N observaciones, tal que $\boldsymbol{y}_k = (y_k^1(\boldsymbol{x}_1; \boldsymbol{w}_k), \ldots, y_k^N(\boldsymbol{x}_N; \boldsymbol{w}_k))^T$, es posible desarrollar cada función de este vector utilizando la combinación lineal de las H funciones base:

$$y_k^n = y_k^n(\boldsymbol{x}_n; \boldsymbol{w}_k) = \sum_{h=1}^{H} R_n^h w_k^h \qquad \forall k, n. \tag{4.21}$$

Si se asume que la distribución a priori de los parámetros $\boldsymbol{w}$ sigue una distribución Gaussiana con media cero y varianza σ_w^2,

$$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{0}, \sigma_w^2 \boldsymbol{I}_w), \tag{4.22}$$

entonces el vector de valores de la función desconocida $\boldsymbol{y}_k$, siendo una función lineal de los parámetros $\boldsymbol{w}$, sigue también una distribución Gaussiana con media cero,

$$p(\boldsymbol{y}_k) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}) \qquad \forall k, \tag{4.23}$$

donde $\boldsymbol{Q}$ representa la matriz de covarianza del vector $\boldsymbol{y}_k$ e $\boldsymbol{I}_w$ en (4.22) corresponde con la matriz identidad del mismo tamaño que el vector de parámetros $\boldsymbol{w}$. Aprovechando el hecho que $\boldsymbol{y}_k = \boldsymbol{R}\boldsymbol{w}_k$, como se definió en (4.21), el valor esperado de $\boldsymbol{y}_k$ es:

$$\langle \boldsymbol{y}_k \rangle = \langle \boldsymbol{R}\boldsymbol{w}_k \rangle = \boldsymbol{R}\langle \boldsymbol{w}_k \rangle \qquad \forall k, \tag{4.24}$$

y por tanto, la matriz de covarianza $\boldsymbol{Q}$ puede definirse como:

$$\boldsymbol{Q} = \langle \boldsymbol{y}_k \boldsymbol{y}_k^T \rangle = \langle \boldsymbol{R}\boldsymbol{w}_k \boldsymbol{w}_k^T \boldsymbol{R}^T \rangle = \boldsymbol{R}\langle \boldsymbol{w}_k \boldsymbol{w}_k^T \rangle \boldsymbol{R}^T \tag{4.25}$$

$$= \sigma_w^2 \boldsymbol{R}\boldsymbol{R}^T, \tag{4.26}$$

de acuerdo a la suposición hecha en (4.22) sobre la distribución a priori de $\boldsymbol{w}$. De esta forma, la distribución a priori del vector $\boldsymbol{y}_k$ en (4.23) puede reescribirse como:

$$p(\boldsymbol{y}_k) = \mathcal{N}(\boldsymbol{0}, \sigma_w^2 \boldsymbol{R}\boldsymbol{R}^T) \qquad \forall k. \tag{4.27}$$

4.2.2 Los procesos Gaussianos como modelos estocásticos

El resultado presentado en (4.27) indica que el vector que contiene las N evaluaciones de las funciones base sigue una distribución Gaussiana [37]. Esto es cierto para cualquier conjunto de observaciones $\boldsymbol{X}_N$, y corresponde con la definición de un proceso Gaussiano (Definición 4.2).

Definición 4.2 *La distribución de probabilidad de una función $y_k(\boldsymbol{x})$ es un proceso Gaussiano si, para una selección finita de observaciones $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, la densidad $p(y_k(\boldsymbol{x}_1), \ldots, y_k(\boldsymbol{x}_N))$ es una distribución Gaussiana.*

Así, un proceso Gaussiano consiste en una distribución de probabilidad sobre funciones. Si se asume que cada elemento en $\boldsymbol{t}^k = (t_1^k, \ldots, t_N^k)^T$ difiere del elemento correspondiente en $\boldsymbol{y}_k$ por un ruido aditivo Gaussiano de varianza σ_v^2 [37], entonces los objetivos $\boldsymbol{t}^k$ tienen también una distribución a priori Gaussiana:

$$p(\boldsymbol{t}^k) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q} + \sigma_v^2 \boldsymbol{I}_N) \qquad \forall k. \tag{4.28}$$

Si se define una matriz $\boldsymbol{C}$ que represente la matriz de covarianza de los N objetivos en (4.28), entonces:

$$\boldsymbol{C} = (\boldsymbol{Q} + \sigma_v^2 \boldsymbol{I}_N) = \sigma_w^2 \boldsymbol{R}\boldsymbol{R}^T + \sigma_v^2 \boldsymbol{I}_N. \tag{4.29}$$

Este resultado permite analizar una nueva perspectiva para el problema de interpolación. En vez de especificar una distribución a priori para las funciones en términos de funciones base y de introducir distribuciones a priori en los parámetros, se puede hacer uso de una función de covarianza [37]. Así, dada una función de covarianza $C(\boldsymbol{x}_n, \boldsymbol{x}_{n'})$ válida es posible redefinir $\boldsymbol{Q}$ en (4.28), donde cada elemento de esta matriz se define como:

$$Q_{nn'} = C(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) \qquad \forall n, n', \tag{4.30}$$

en donde los subíndices n y n' representan el par de observaciones en que se evalúa la función de covarianza, pudiendo darse el caso $n = n'$. Bajo la suposición de que $\boldsymbol{y}_k$ difiere de $\boldsymbol{t}^k$ por un ruido aditivo Gaussiano de varianza σ_v^2, cada elemento de la matriz de covarianza $\boldsymbol{C}$ se determina mediante:

$$C_{nn'} = Q_{nn'} + {\sigma_v}^2 \delta_{nn'} = C(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) + {\sigma_v}^2 \delta_{nn'} \qquad \forall n, n', \tag{4.31}$$

donde $\delta_{nn'}$ representa una delta de Dirac.

Del mismo modo en que la distribución Gaussiana multivariada está completamente definida por un vector de medias, $\boldsymbol{\mu}$, y una matriz de covarianza, $\boldsymbol{\Sigma}$, un proceso Gaussiano está completamente definido por una función de media, que a lo largo de esta tesis se define como la función cero, y la selección de una función para formar la matriz de covarianza $\boldsymbol{C}$ en (4.29) [42]. De esta forma, tomar una muestra de un proceso Gaussiano implica muestrear una función aleatoria con distribución $\mathcal{GP}(\boldsymbol{0}, \boldsymbol{C})$, donde $\mathcal{GP}$ indica un proceso Gaussiano. Los procesos Gaussianos promedian entonces sobre todas las posibles funciones aleatorias que pueden representar las observaciones. Por su parte, la distribución a priori de los N objetivos en $\boldsymbol{t}^k$ es Gaussiana con matriz de covarianza $\boldsymbol{C}$,

$$p(\boldsymbol{t}^k) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}) = \frac{1}{(2\pi)^{u/2}|\boldsymbol{C}|^{1/2}} \exp\left[-\frac{(\boldsymbol{t}^k)^T \boldsymbol{C}^{-1} \boldsymbol{t}^k}{2}\right] \qquad \forall k. \tag{4.32}$$

Esto indica que definiendo una función *válida* de covarianza y utilizándola para formar $\boldsymbol{C}$ se puede realizar inferencia sobre un nuevo objetivo t_f^k dado el vector observado $\boldsymbol{t}^k$. La validez de la función de covarianza es un tema que se trata en la Sección 4.2.4.

4.2.3 Los procesos Gaussianos como modelos de predicción

Sea $\boldsymbol{x}_f$ una observación de prueba para la cual se desean inferir los objetivos $\boldsymbol{t}_f$. Una vez que se ha formado la matriz de covarianza $\boldsymbol{C}$ definida en (4.29) es posible inferir el objetivo t_f^k dados los objetivos de entrenamiento $\boldsymbol{t}^k$ mediante la densidad conjunta $p(t_f^k, \boldsymbol{t}^k)$ [53]. De acuerdo a la definición de un proceso Gaussiano (Definición 4.2), esta densidad de probabilidad conjunta sigue una distribución Gaussiana con media cero y matriz de covarianza $\boldsymbol{C}_F$:

$$p(t_f^k, \boldsymbol{t}^k) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}_F) \qquad \forall k, \tag{4.33}$$

donde $\boldsymbol{C}_F$ es la matriz de covarianza del vector $\boldsymbol{t}_F^k = (t_1^k, \ldots, t_N^k, t_f^k)^T$, con dimensión $(N+1) \times (N+1)$. Así, la distribución condicional de interés $p(t_f^k | \boldsymbol{t}^k)$ tiene también una distribución Gaussiana [53]. Si se definen submatrices en $\boldsymbol{C}_F$ de tal forma que $\boldsymbol{C}_N$ sea la matriz de covarianza evaluada en las observaciones $\boldsymbol{X}_N$, tal que $\boldsymbol{C}_N = C(\boldsymbol{X}_N, \boldsymbol{X}_N)$; que $\boldsymbol{b}_1$ sea el vector de covarianza evaluada tanto en las observaciones $\boldsymbol{X}_N$ como en la observación de prueba $\boldsymbol{x}_f$, tal que $\boldsymbol{b}_1 = C(\boldsymbol{X}_N, \boldsymbol{x}_f)$; y que v_1 sea la covarianza evaluada en la observación de prueba $\boldsymbol{x}_f$, tal que $v_1 = C(\boldsymbol{x}_f, \boldsymbol{x}_f)$, entonces $\boldsymbol{C}_F$ puede reescribirse como:

$$\Big[\boldsymbol{C}_F\Big] = \begin{bmatrix} [\boldsymbol{C}_N] & [\boldsymbol{b}_1] \\ \\ [\boldsymbol{b}_1]^T & [v_1] \end{bmatrix}, \tag{4.34}$$

y la distribución condicional de interés es:

$$p(t_f^k|\boldsymbol{t}^k) = \frac{p(t_f^k, \boldsymbol{t}^k)}{p(\boldsymbol{t}^k)} \propto \exp\left[-\frac{(\boldsymbol{t}_F^k)^T \boldsymbol{C}_F^{-1} \boldsymbol{t}_F^k}{2}\right] \qquad \forall k. \tag{4.35}$$

A pesar de que es posible evaluar tanto la media como la varianza de la distribución posterior de t_f^k por la inversión directa de $\boldsymbol{C}_F$ en la ecuación (4.35), la complejidad computacional de tal inversión es $(N+1)^3$, por lo que este procedimiento se vuelve computacionalmente costoso conforme crece el tamaño del conjunto de observaciones $\boldsymbol{X}_N$ y conforme se tienen más observaciones de prueba. Para evitar esto, McKay [37] sugiere utilizar las ecuaciones de particiones inversas [6] para reescribir $\boldsymbol{C}_F$ en términos de la matriz de covarianza de las N observaciones, $\boldsymbol{C}_N$. Utilizando esta técnica, se reescribe la inversa de la matriz de covarianza en (4.34) como:

$$\Big[\boldsymbol{C}_F\Big]^{-1} = \begin{bmatrix} [\boldsymbol{B}] & [\boldsymbol{b}_2] \\ \\ [\boldsymbol{b}_2]^T & [v_2] \end{bmatrix}, \tag{4.36}$$

donde:

$$v_2 = (v_1 - \boldsymbol{b}_1^T \boldsymbol{C}_N^{-1} \boldsymbol{b}_1)^{-1}, \tag{4.37}$$

$$\boldsymbol{b}_2 = -v_2 \boldsymbol{C}_N^{-1} \boldsymbol{b}_1, \tag{4.38}$$

$$\boldsymbol{B} = \boldsymbol{C}_N^{-1} + \frac{1}{v_2}\left[\boldsymbol{b}_2 \boldsymbol{b}_2^T\right]. \tag{4.39}$$

Aplicando estas ecuaciones en la distribución posterior desarrollada en (4.33) se obtiene una formulación para evaluar la densidad condicional $p(t_f^k, \boldsymbol{t}^k)$ en función de la inversa de la matriz de covarianza de los datos de entrenamiento C_N:

$$p(t_f^k|\boldsymbol{t}^k) = \mathcal{N}(\widehat{t}_f^{\,k}, \boldsymbol{\Sigma}_{\widehat{t}_f^{\,k}}) \qquad \forall k, \tag{4.40}$$

en donde:

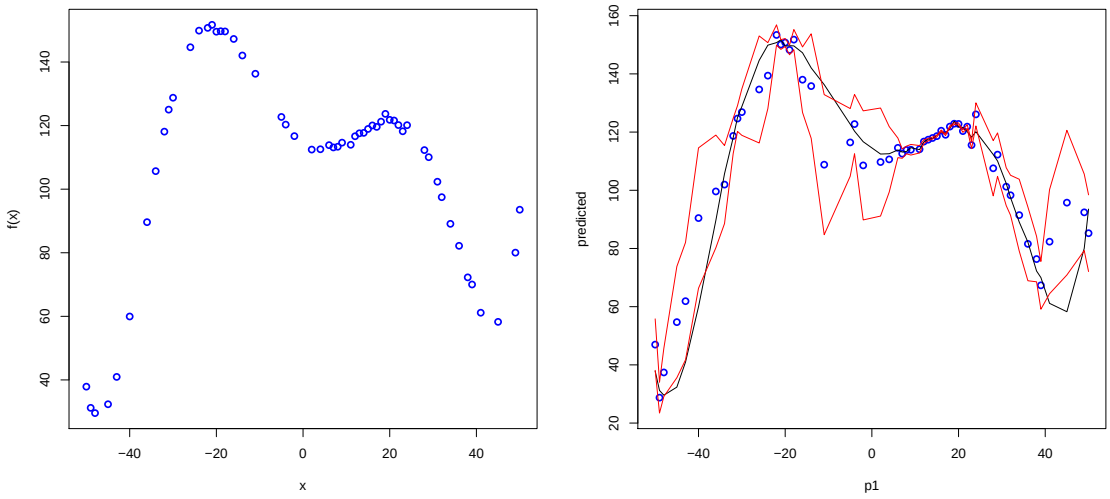
$$\widehat{t}_f^{\,k} = \boldsymbol{b}_1^T \boldsymbol{C}_N^{-1} \boldsymbol{t}^k \qquad \forall k, \tag{4.41}$$

$$\boldsymbol{\Sigma}_{\widehat{t}_f^{\,k}} = v_1 - \boldsymbol{b}_1^T \boldsymbol{C}_N^{-1} \boldsymbol{b}_1 \qquad \forall k. \tag{4.42}$$

De esta forma, el valor esperado del objetivo t_f^k dada la observación $\boldsymbol{x}_f$ está dada por $\widehat{t}_f^{\,k}$, mientras que $\Sigma^2_{\widehat{t}_f^{\,k}}$ define el margen potencial de error de esta predicción, por lo que es posible cuantificar la precisión del valor inferido [42]. Esto representa una ventaja al utilizar procesos Gaussianos en problemas de inferencia, por encima de un enfoque paramétrico como lo son las redes neuronales. Es importante notar que no es necesario invertir la matriz $\boldsymbol{C}_F$ para hacer predicciones en $\boldsymbol{x}_f$, sólo $\boldsymbol{C}_N$. Así, los procesos Gaussianos permiten la implementación de un modelo con un número H de funciones base más grande que el número de datos N, siendo la complejidad computacional de orden N^3, independientemente del valor de H y del tamaño del conjunto de prueba.

En la Figura 4.3 se muestra la predicción de un conjunto de observaciones mediante el entrenamiento de un proceso Gaussiano[1]. Una de las ventajas de modelar con procesos Gaussianos es la posibilidad de cuantificar el error en cada punto pronosticado, como se observa en 4.3b. A pesar de que existe ruido en las observaciones, conforme aumenta la concentración de puntos de entrenamiento dentro de un intervalo la variabilidad de las predicciones dentro del mismo disminuye, ocasionando una disminución en el error de predicción, como sucede cuando $x \approx 20$ (x aparece renombrado como $p1$ en esta figura). Sin embargo, cuando se tiene una concentración baja de puntos de entrenamiento el efecto que produce el ruido impide que el

[1]El entrenamiento de este proceso Gaussiano se realizó utilizando el paquete *mlegp* del proyecto *R Statistical Software* [50], debido a la facilidad que ofrece para graficar los resultados



(a) Conjunto de observaciones.

(b) Predicción del proceso Gaussiano.

Figura 4.3: Predicción de un proceso Gaussiano entrenado con un conjunto trivial de observaciones, en donde $f(x)$ corresponde con la función polinomial de sexto orden $f(x) = 1151 - 10x + x^2 + 7.2 \times 10^{-3}x^3 - 1.5 \times 10^{-3}x^4 - 4 \times 10^{-07}x^5 + 4 \times 10^{-07}x^6$ con un ruido aditivo normal estándar, $\mathcal{N}(0, 1)$. En (a) se presenta el conjunto de observaciones, mientras que (b) muestra en azul la predicción del proceso Gaussiano con la función exponencial cuadrada como función de covarianza. Las líneas en rojo representan el margen potencial de error de la predicción, y la línea en negro la posición de las observaciones del conjunto de entrenamiento.

modelo estime adecuadamente la función detrás de las observaciones, lo que produce un incremento en el error de predicción del modelo, como puede observarse cuando $x \approx -5$ y cuando $x \approx 50$.

4.2.4 La función de covarianza como componente de los procesos Gaussianos

Como se ha visto hasta el momento, las predicciones hechas a través de los procesos Gaussianos en esta tesis dependen completamente de la matriz de covarianza $\boldsymbol{C}$. En los inicios de esta sección se precisó que era necesario definir una función de covarianza *válida* para construir una matriz de covarianza capaz de realizar predicciones. La única restricción en la selección de esta función de covarianza es que debe generar una matriz de covarianza no-negativa definida para cualquier conjunto de puntos $\{\boldsymbol{x}_n\}_{n=1}^{N}$. Para una explicación detallada y concisa de lo que se considera una función de covarianza válida véase [37]. Existen dos tipos de funciones de covarianza que son comúnmente aplicadas en la literatura de los procesos Gaussianos: las estacionarias y las no-estacionarias. Las funciones de covarianza estacionarias están únicamente en función de la distancia entre observaciones, sin tomar en cuenta algún tipo de patrón periódico en las observaciones. La ideología detrás de este tipo de funciones de covarianza implica que las observaciones que son semejantes a una observación bajo análisis proporcionan información acerca de la inferencia para tal observación. Un ejemplo ilustrativo de esta situación corresponde con la estimación de la afinidad en acoplamientos enzimáticos: al analizar la interacción entre una enzima y un sustrato, el resto de las interacciones entre enzimas y sustratos semejantes aportarán información acerca de la energía de interacción del par bajo estudio, dada la gran selectividad que tienen los sitios de interacción en los complejos proteicos. Por su parte, las funciones de covarianza no estacionarias tienen la habilidad de detectar patrones periódicos encontrados en las observaciones en adición a las propiedades de las funciones no-estacionarias.

Para esta tesis se considera el uso de la función exponencial cuadrada (ec. 4.43)

como función de covarianza. La función exponencial cuadrada pertenece al conjunto de funciones estacionarias, dado que los problemas de estudio que se presentan en esta tesis se consideran invariantes en el tiempo, y ha sido seleccionada de entre las funciones estacionarias debido a que su aplicación en los problemas de estudio propuestos en esta tesis ha mostrado tener buenos resultados, además de ser una de las funciones de covarianza más estudiadas.

$$C(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) = \sigma_f^2 \exp\left[-\frac{1}{2}\sum_{i=1}^{p}\frac{(x_n^i - x_{n'}^i)^2}{l_i^2}\right] \qquad \forall n, n'. \tag{4.43}$$

Recordando que se ha realizado la suposición en que $\boldsymbol{y}_k$ difiere de $\boldsymbol{t}^k$ por un ruido aditivo Gaussiano de varianza σ_v^2, la función de covarianza resultante corresponde con:

$$C(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) = \sigma_f^2 \exp\left[-\frac{1}{2}\sum_{i=1}^{p}\frac{(x_n^i - x_{n'}^i)^2}{l_i^2}\right] + \delta\sigma_v^2 \qquad \forall n, n', \tag{4.44}$$

en donde σ_f^2 es la varianza de la señal, la cual controla el escalamiento vertical de la función; σ_v^2 es la varianza del proceso asumido para el ruido y l_i son hiperparámetros conocidos como longitudes características que están asociados a la dimensión i de las observaciones. Un valor grande de l_i indica que las funciones $\boldsymbol{y}(\boldsymbol{x}_n)$ serán prácticamente funciones constantes de $\boldsymbol{x}^i = (x_1^i, \ldots, x_N^i)^T$. Por su parte, δ representa una delta de Dirac.

4.2.5 Aprendizaje en los procesos Gaussianos

Una vez que se ha seleccionado y definido la función de covarianza es necesario aprender los hiperparámetros $\boldsymbol{\theta} = (\sigma_f^2, \sigma_v^2, l_1, \ldots, l_p)^T$ que la componen a partir de las observaciones disponibles. La distribución posterior de los hiperparámetros de la función de covarianza dado el conjunto de observaciones es:

$$p(\boldsymbol{\theta}|\boldsymbol{X}_N, \boldsymbol{t}^k) \propto p(\boldsymbol{t}^k|\boldsymbol{\theta}, \boldsymbol{X}_N)p(\boldsymbol{\theta}). \tag{4.45}$$

El primer término corresponde con la verosimilitud de los hiperparámetros, es decir, su evidencia. A partir de la suposición para el proceso de error,

$$\begin{aligned} p(\boldsymbol{t}^k|\boldsymbol{\theta}, \boldsymbol{X}_N) &= \prod_{k=1}^{u} \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}_N) \\ &= \frac{1}{(2\pi)^{Nu/2}|\boldsymbol{C}_N|^{u/2}} \exp\left[-\frac{1}{2}\sum_{k=1}^{u}(\boldsymbol{t}^k)^T\boldsymbol{C}_N^{-1}\boldsymbol{t}^k\right]. \end{aligned} \tag{4.46}$$

De nueva cuenta, se evalúa la función logarítmica de la verosimilitud en (4.46) por ser una función numéricamente más conveniente para propósitos de optimización:

$$\ln p(\boldsymbol{t}^k|\boldsymbol{\theta}, \boldsymbol{X}_N) = -\frac{u}{2}\ln|\boldsymbol{C}_N| - -\frac{1}{2}\sum_{k=1}^{u}(\boldsymbol{t}^k)^T\boldsymbol{C}_N^{-1}\boldsymbol{t}^k - \frac{Nu}{2}\ln(2\pi). \tag{4.47}$$

Al ser la función logarítmica una función monótona creciente, ésta alcanza su máximo en el mismo punto en que lo hace la función original. Esta es una propiedad importante de la función logarítmica cuando se busca el conjunto de parámetros que maximizan la función de verosimilitud.

Idealmente, para realizar predicciones en un enfoque Bayesiano se define una distribución a priori para los hiperparámetros y en conjunto con la función de verosimilitud se realiza una marginalización sobre ellos. La marginalización sobre los hiperparámetros, $\boldsymbol{\theta}$, del modelo corresponde con:

$$p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{X}_N, \boldsymbol{T}_N) = \int p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{\theta}, \boldsymbol{X}_N, \boldsymbol{T}_N)p(\boldsymbol{\theta}|\boldsymbol{X}_N, \boldsymbol{T}_N)d\boldsymbol{\theta}. \tag{4.48}$$

Sin embargo, esta integral es intratable [37]. Esto conduce los cálculos hacia dos caminos posibles. El primero de ellos corresponde con realizar una aproximación

numérica a la integral usando un método tipo Monte Carlo, de forma similar al procedimiento empleado en el caso de las redes neuronales. El segundo camino es aproximar la integral usando los valores más probables de los hiperparámetros, dado que el estimador posterior máximo de $\boldsymbol{\theta}$ ocurre cuando la distribución condicional $p(\boldsymbol{\theta}|\boldsymbol{X}_N, \boldsymbol{T}_N)$ se encuentra en su punto máximo. Sean $\boldsymbol{\theta}_{MP}$ el conjunto de hiperparámetros más probables, tal aproximación corresponde con:

$$p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{X}_N, \boldsymbol{T}_N) \simeq p(\boldsymbol{t}_f|\boldsymbol{x}_f, \boldsymbol{X}_N, \boldsymbol{T}_N, \boldsymbol{\theta_{MP}}). \tag{4.49}$$

La aproximación en (4.49) es la opción seleccionada en esta tesis. El teorema de Bayes (Teorema 2.1) indica que cuando no se tiene suficiente información acerca de los hiperparámetros $\boldsymbol{\theta}$, el punto máximo de la distribución condicional $p(\boldsymbol{\theta}|\boldsymbol{X}_N, \boldsymbol{T}_N)$, y por ende el conjunto óptimo de hiperparámetros, corresponde con la maximización de $\ln p(\boldsymbol{t}^k|\boldsymbol{\theta}, \boldsymbol{X}_N)$ en (4.47). A pesar de utilizar el conjunto óptimo de hiperparámetros en vez de computar el valor esperado mediante sus respectivas distribuciones posteriores, esta elección presenta buenos resultados al aplicarla en los procesos Gaussianos como método de solución para los problemas de estudio propuestos en esta tesis, además de presentar ventajas importantes con respecto al tiempo de cómputo. Sin embargo, hay una cuestión que debe ser tomada en cuenta al elegir esta alternativa: la función de verosimilitud marginal en (4.47) puede ser multimodal, por lo que se necesita un método de optimización global que sea capaz de hacer una búsqueda eficiente del espacio de soluciones para estimar el conjunto óptimo de hiperparámetros. Aunado a esto, la función a estimar en la predicción de la afinidad en acoplamientos enzimáticos (Sección 3.2) es altamente no-diferenciable. Esto implica que los algoritmos de optimización basados en gradientes no puedan ser aplicados para la optimización de los hiperparámetros en este problema de estudio. Una rutina de optimización útil para lidiar con estos dos obstáculos es el recocido simulado, introducido en la Sección 2.3 y seleccionado en esta tesis como rutina de optimización. El algoritmo del recocido simulado es capaz de realizar una búsqueda eficiente del óptimo global en el espacio de soluciones, además de no necesitar la evaluación del

gradiente.

4.3 Mezcla Infinita de Gaussianas

Un modelo finito de mezcla de Gaussianas es una densidad paramétrica de probabilidad representada como una suma ponderada de componentes con distribución Gaussiana, por lo que se trata de una técnica efectiva que tiene sus raíces en la literatura estadística [39]. En años recientes, este método ha sido una herramienta aplicada en el área del aprendizaje computacional para resolver problemas provenientes de diferentes áreas del conocimiento [9], debido a ciertas propiedades y ventajas que exhibe sobre otros métodos, las cuales son abordadas en esta sección. Un modelo de mezcla de Gaussianas asume que el conjunto de observaciones $\mathcal{D}$ es generado por una mezcla de k Gaussianas multivariadas:

$$p(\mathcal{D}|\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, \boldsymbol{S}_1, \ldots, \boldsymbol{S}_k, \pi_1, \ldots, \pi_k) = \sum_{j=1}^{k} \pi_j \, \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{S}_j^{-1}), \tag{4.50}$$

en donde las observaciones $\mathcal{D}$ incluyen tanto los atributos de entrada como los objetivos, tal que $\mathcal{D} = \{\boldsymbol{x}_n, \boldsymbol{t}_n\}_{n=1}^{N}$. Nótese que en este conjunto de entrenamiento se incluyen todos los objetivos en $\boldsymbol{t}_n$, a diferencia de las redes neuronales y los procesos Gaussianos recién descritos, en donde se formaban k conjuntos de entrenamiento, cada uno con un objetivo t_n^k. Por su parte, $\boldsymbol{\mu}_j$ es el vector de medias de la Gaussiana j, $\boldsymbol{S}_j$ es la matriz de precisiones (o covarianza inversa) del componente j y π_j es la proporción de la clase[2] j en la mezcla. Una ventaja de aplicar el modelo de mezcla de Gaussianas para problemas de inferencia es que no existe una distinción fija entre entradas y salidas (i.e., entre observaciones y objetivos) durante la etapa de aprendizaje, por lo que una vez que el modelo es determinado se puede especificar cualquier subconjunto dimensional de entradas y salidas, y computar el valor esperado de las dimensiones restantes.

[2]En esta tesis se utilizan indistintamente los términos clase y componente para indicar una Gaussiana perteneciente al modelo.

En la Figura 4.4 se muestran los resultados al entrenar un modelo de mezcla finita de Gaussianas[3] para estimar la distribución de las observaciones. Son tres los componentes que modelan tal distribución, representados por elipses. Las líneas punteadas dentro de cada elipse corresponden con la varianza individual de los dos atributos, mientras que su intersección corresponde con la media de cada componente. La coloración de las observaciones indica los subconjuntos asociados a cada componente. Así, condicionando el atributo x_2 con x_1 mediante la distribución $p(x_2|x_1)$ se genera un modelo de inferencia, en donde el valor esperado para el atributo x_2 se determina a partir de los atributos x_1. Esta transformación se abordará más delante.

En la versión Bayesiana de un modelo de mezcla de Gaussianas no es necesario limitar de antemano el número de componentes, k, en la mezcla para que éste tome un valor finito, lo que representa una ventaja sobre la versión frecuentista del método. En el área del aprendizaje Bayesiano para mezclas de Gaussianas, un método probabilístico de gran interés es el modelo de mezclas infinitas de Gaussianas, desarrollado por Rasmussen [52]. La ventaja más importante de este algoritmo es su capacidad para inferir automáticamente un número adecuado de componentes en la mezcla, lo que representa una gran problemática en la versión frecuentista del mismo. El resto de esta sección se dedica a introducir los parámetros e hiperparámetros del modelo de mezcla finita de Gaussianas, sus distribuciones a priori y el desarrollo de las distribuciones posteriores para el caso univariado (Sección 4.3.1). Posteriormente se analiza la variación de tales distribuciones cuando el número de componentes k tiende a infinito (Sección 4.3.2) y se generalizan para el caso multivariado (Sección 4.3.3). Una vez que las distribuciones posteriores de los parámetros e hiperparámetros han sido desarrolladas y analizadas, se desarrollan las distribuciones condicionales necesarias para utilizar el modelo de mezcla infinita de Gaussianas como método de predicción (Sección 4.3.4).

[3]El entrenamiento de este modelo de mezcla finita de Gaussianas se realizó utilizando el paquete *mclust* del proyecto *R Statistical Software* [50]

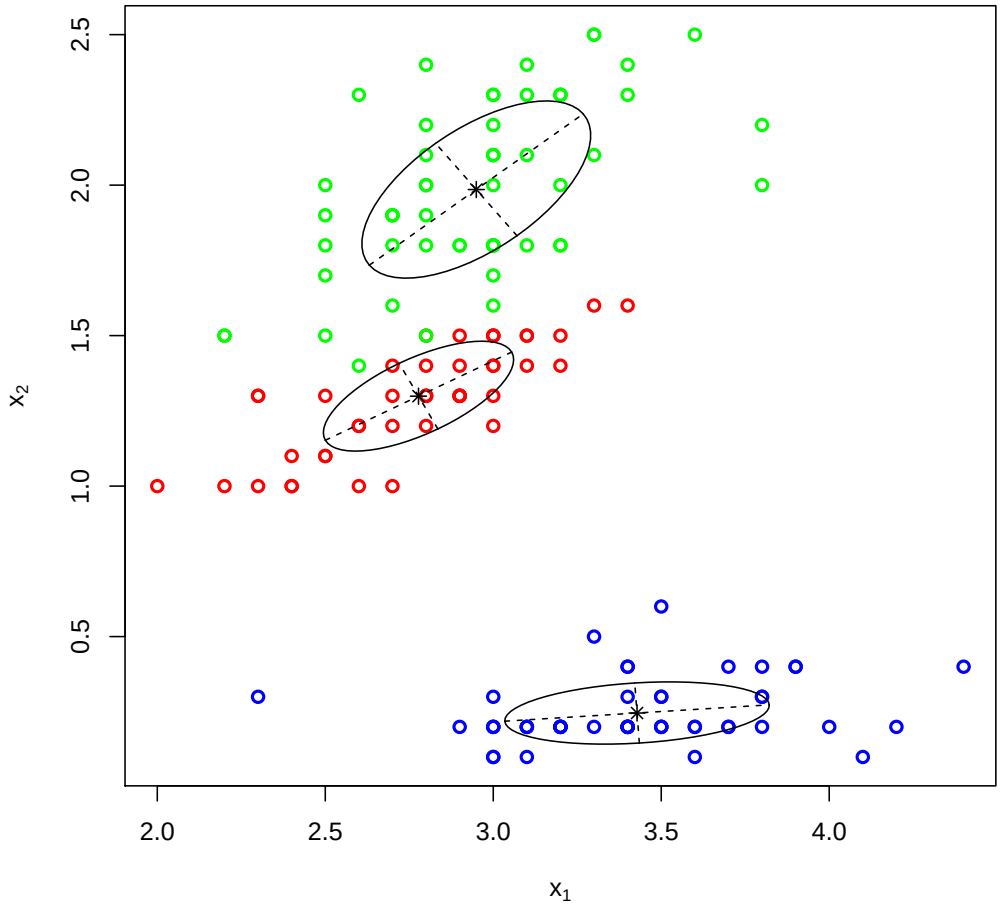


Figura 4.4: Agrupamiento para un conjunto de observaciones bidimensionales mediante un modelo de mezcla finita de Gaussianas. Las elipses representan los tres componentes Gaussianos que conforman la distribución de las observaciones. Las líneas punteadas dentro de cada elipse denotan la varianza para cada atributo, mientras que su intersección representa la media del componente. En azul, rojo y verde se representan los tres subconjuntos de observaciones por los que cada componente se hace responsable.

4.3.1 Parámetros e hiperparámetros de la mezcla de Gaussianas

El modelo de mezclas infinitas de Gaussianas es un modelo Bayesiano jerárquico que implica muestrear de la distribución posterior de un modelo ordinario de mezcla de Gaussianas pero con un número posiblemente infinito de componentes, dadas las observaciones $\mathcal{D}$, las cuales son consideradas observaciones univariadas en esta sección para fines ilustrativos, de modo que $\mathcal{D} = \{y_n\}_{n=1}^N$. Los parámetros de las distribuciones Gaussianas que componen la mezcla, como lo son las medias, $\{\mu_j\}_{j=1}^k$, las precisiones, $\{s_j\}_{j=1}^k$, y las proporciones, $\{\pi_j\}_{j=1}^k$, conforman el conjunto de parámetros de la primera fase (i.e., aquellos asociados directamente al modelo).

Las medias de los componentes y sus hiperparámetros. A las medias de tales Gaussianas se les asignan distribuciones Gaussianas a priori:

$$p(\mu_j|\lambda, r) \sim \mathcal{N}(\lambda, r^{-1}) \qquad \forall j, \tag{4.51}$$

en donde la media, λ, y la precisión, r, son hiperparámetros de una segunda fase y son comunes para todos los componentes. A su vez, a estos hiperparámetros se les asignan distribuciones Gaussianas y gamma a priori, respectivamente:

$$p(\lambda) \sim \mathcal{N}(\mu_{\mathcal{D}}, \sigma_{\mathcal{D}}^2), \tag{4.52}$$

$$p(r) \sim \mathcal{G}(1, \sigma_{\mathcal{D}}^{-2}) \propto r^{-1/2} \exp\left[-r\sigma_{\mathcal{D}}^2/2\right], \tag{4.53}$$

en donde $\mu_{\mathcal{D}}$ y $\sigma_{\mathcal{D}}^2$ corresponden con la media y la varianza, respectivamente, del conjunto de observaciones $\mathcal{D}$. Al parámetro de forma de la distribución gamma en (4.53) se le asigna la unidad, lo que implica una distribución vaga (i.e., no se aporta suficiente información sobre la distribución a priori de este hiperparámetro).

Ahora bien, si se introduce una variable estocástica, $\boldsymbol{c} = \{c_n\}_{n=1}^{N}$, para cada observación en $\mathcal{D}$ que actúe como indicador del componente que ha generado la observación, es posible obtener la distribución posterior para las medias de los componentes multiplicando la función de verosimilitud del modelo (ec. 4.50), condicionada en los indicadores, por su distribución a priori (ec. 4.51). De esta forma:

$$p(\mu_j|\boldsymbol{c}, \mathcal{D}, s_j, \lambda, r) \sim \mathcal{N}\left(\frac{\bar{y}_j h_j s_j + \lambda r}{h_j s_j + r}, \frac{1}{h_j s_j + r}\right) \qquad \forall j. \tag{4.54}$$

A la variable h_j se le denomina el número de ocupación, y representa la cantidad de observaciones que pertenecen a la clase j, mientras que $\bar{y}_j$ representa la media de tales observaciones:

$$\bar{y}_j = \frac{1}{h_j} \sum_{i:c_n=j} y_i. \tag{4.55}$$

Para los hiperparámetros de las medias de los componentes (i.e., para λ y r), la distribución Gaussiana a priori de la media (ec. 4.51) asume el papel de la función de verosimilitud, que junto con la distribución a priori anterior (ec. 4.54), producen distribuciones posteriores de forma estándar:

$$p(\lambda|\mu_1, \ldots, \mu_k, r) \sim \mathcal{N}\left(\frac{\mu_{\mathcal{D}}\sigma_{\mathcal{D}}^{-2} + r\sum_{j=1}^{k}\mu_j}{\sigma_{\mathcal{D}}^{-2} + kr}, \frac{1}{\sigma_{\mathcal{D}}^{-2} + kr}\right), \tag{4.56}$$

$$p(r|\mu_1, \ldots, \mu_k, \lambda) \sim \mathcal{G}\left(k+1, \left[\frac{1}{k+1}\left(\sigma_{\mathcal{D}}^2 + \sum_{j=1}^{k}(\mu_j - \lambda)^2\right)\right]^{-1}\right). \tag{4.57}$$

Las precisiones de los componentes y sus hiperparámetros. Retornando a los parámetros de la primera fase, a las precisiones de las Gaussianas que conforman la mezcla, s_j, se les asigna una distribución gamma a priori:

$$p(s_j|\beta, w) \sim \mathcal{G}(\beta, w^{-1}) \qquad \forall j, \tag{4.58}$$

en donde el parámetro de forma, β, y la media, w^{-1}, son hiperparámetros de la segunda fase, y son comunes para todos los componentes. De nueva cuenta, a estos hiperparámetros se les asignan distribuciones gamma inversa y gamma como distribuciones a priori, respectivamente:

$$p(\beta^{-1}) \sim \mathcal{G}(1,1) \Longrightarrow p(\beta) \propto \beta^{-3/2} \exp\left[-1/(2\beta)\right], \tag{4.59}$$

$$p(w) \sim \mathcal{G}(1, \sigma_{\mathcal{D}}^2). \tag{4.60}$$

La distribución posterior de las precisiones de los componentes se obtiene multiplicando la función de verosimilitud del modelo (ec. 4.50) condicionada en los indicadores, $\boldsymbol{c}$, por su distribución a priori (ec. 4.58):

$$p(s_j|\boldsymbol{c}, \mathcal{D}, \mu_j, \beta, w) \sim \mathcal{G}\left(\beta + h_j, \left[\frac{1}{\beta + h_j}\left(w\beta + \sum_{i:c_i=j}(y_i - \mu_j)^2\right)\right]^{-1}\right) \quad \forall j. \tag{4.61}$$

Para los hiperparámetros de las precisiones de los componentes (i.e., para w y β), la distribución a priori de las precisiones (ec. 4.58) asume el papel de la función de verosimilitud, que junto con las distribuciones a priori para éstos (ec. 4.59 y ec. 4.60) proporcionan:

$$p(w|s_1, \ldots, s_k, \beta) \sim \mathcal{G}\left(k\beta + 1, \left[\frac{1}{k\beta + 1}\left(\sigma_{\mathcal{D}}^{-2} + \beta\sum_{j=1}^{k} s_j\right)\right]\right), \tag{4.62}$$

$$p(\beta|s_1, \ldots, s_k, w) \propto \Gamma\left(\frac{\beta}{2}\right)^{-k} \exp\left(\frac{-1}{2\beta}\right)\left(\frac{\beta}{2}\right)^{(k\beta-3)/2}$$

$$\times \prod_{j=1}^{k} (s_j w)^{\beta/2} \exp\left[-\frac{\beta s_j w}{2}\right]. \tag{4.63}$$

La distribución posterior de β en (4.63) no es de forma estándar, pero es posible demostrar que $p(\ln(\beta)|s_1, \ldots, s_k, w)$ es una función logarítmica cóncava, de modo que es posible generar muestras independientes de $\ln(\beta)$ utilizando el muestreo de Gibbs con paso Metrópolis y aplicando una transformación para obtener muestras provenientes de la distribución posterior de β. La distribución logarítmica posterior de β corresponde con:

$$\begin{aligned} \ln p(\beta|s_1, \ldots, s_k) &= -k \ln \Gamma\left(\frac{\beta}{2}\right) - \frac{1}{2\beta} + \frac{k\beta - 3}{2} \ln\left(\frac{\beta}{2}\right) \\ &\quad + \sum_{j=1}^{k} \left(\frac{\beta}{2}\right) (\ln s_j + \ln w) - \frac{\beta s_j w}{2}. \end{aligned} \tag{4.64}$$

Las proporciones de la mezcla. Retornando una vez mas a los parámetros de la primera fase, a las proporciones de la mezcla, π_j, se les asigna una distribución Dirichlet a priori, con un parámetro de concentración de α/k:

$$p(\pi_1, \ldots, \pi_k|\alpha) \sim Dirichlet(\alpha/k, \ldots, \alpha/k) = \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k} \prod_{j=1}^{k} \pi_j^{\alpha/k-1}. \tag{4.65}$$

Nótese que las proporciones deben ser positivas y sumar la unidad:

$$\sum_{j=1}^{k} \pi_j = 1. \tag{4.66}$$

Dadas las proporciones de la mezcla, la distribución a priori para los parámetros de ocupación, h_j, es multinomial, mientras que la distribución conjunta de los indicadores se vuelve:

$$p(c_1, \ldots, c_N|\pi_1, \ldots, \pi_k) = \prod_{j=1}^{k} \pi_j^{h_j}, \tag{4.67}$$

en donde:

$$h_j = \sum_{n=1}^{N} \delta_{\text{Kroenecker}}(c_n, j). \tag{4.68}$$

Utilizando la integral Dirichlet estándar es posible integrar la proporción de la mezcla y escribir la distribución a priori directamente en términos de los indicadores [52]:

$$\begin{aligned} p(c_1, \ldots, c_N|\alpha) &= \int p(c_1, \ldots, c_N|\pi_1, \ldots, \pi_k) p(\pi_1, \ldots, \pi_k) d\pi_1 \ldots d\pi_k \\ &= \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k} - \prod_{j=1}^{k} \pi_j^{h_j+\alpha/k-1} d\pi_j = \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k} \prod_{j=1}^{k} \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k}. \end{aligned} \tag{4.69}$$

Para poder aplicar el muestreo de Gibbs a los indicadores, $\boldsymbol{c}$, es necesario determinar la distribución condicional a priori para un indicador dado el resto de ellos. Esta distribución se obtiene a partir de la distribución a priori de los indicadores (ec. 4.69) manteniendo todos los indicadores fijos, con excepción de uno:

$$p(c_n = j|c_{n'}, \alpha) = \frac{h_{n',j} + \alpha/k}{N - 1 + \alpha} \qquad \forall n, \tag{4.70}$$

donde n' indica todos los índices con excepción de n y $h_{n',j}$ es la cantidad de observaciones, excluyendo y_n, que están asociadas al componente j. Finalmente, se asigna una distribución gamma inversa en los parámetros de concentración, α:

$$p(\alpha^{-1}) \sim \mathcal{G}(1, 1) \Longrightarrow p(\alpha) \propto \alpha^{-3/2} \exp\left[-1/(2\alpha)\right]. \tag{4.71}$$

La función de verosimilitud para α se deriva a partir de la distribución a priori de los indicadores (ec. 4.69), que junto con la distribución a priori para α (ec. 4.71) produce:

$$p(h_1, \ldots, h_k|\alpha) = \frac{\alpha^k \Gamma(\alpha)}{\Gamma(N + \alpha)}, \tag{4.72}$$

$$p(\alpha|k,N) \propto \frac{\alpha^{k-3/2}\exp\left[-1/(2\alpha)\right]\Gamma(\alpha)}{\Gamma(N+\alpha)}. \tag{4.73}$$

Nótese que la distribución posterior de α depende sólo del número de observaciones, N, y del número de componentes, k, y no de la forma en que las observaciones se distribuyen entre los componentes. La distribución $\ln p(\alpha|k,N)$ es una función logarítmica cóncava, de tal forma que se puede aplicar el muestreo de Gibbs para generar muestras independientes de esta distribución. La distribución logarítmica para α se define mediante:

$$\ln p(\alpha|k,N) = (k-3/2)\ln\alpha - \frac{1}{2\alpha}\ln\Gamma(\alpha) - \ln\left(\Gamma(N+\alpha)\right). \tag{4.74}$$

En la siguiente sección se generaliza este método para el caso en que se tiene una cantidad infinita de componentes.

4.3.2 El límite infinito en la mezcla de Gaussianas

Hasta este punto se ha considerado a k como una cantidad fija y finita. De acuerdo a Rasmussen [52], para todas las variables del modelo, con excepción de los indicadores $\boldsymbol{c}$, las distribuciones posteriores de probabilidad para el límite infinito se obtienen sustituyendo k por el número de clases que tienen al menos una observación asociada, k_{rep}, en las distribuciones posteriores derivadas para el modelo finito. Así, k_{rep} indica las clases que están representadas en el modelo e indica el número de componentes que éste tiene. En el caso de los indicadores, dejando que $k \to \infty$ en (4.70), las distribuciones a priori tienden a diferentes límites de acuerdo al número de observaciones asociadas a los componentes. Las clases que tienen al menos una observación asociada diferente a y_n, es decir, las clases que satisfacen $h_{n',j} > 0$, tienden a:

$$p(c_n = j|c_{n'},\alpha) = \frac{h_{n',j}}{N-1+\alpha}. \tag{4.75}$$

Por otro lado, la combinación de todos los componentes con excepción de aquellos en que $h_{i',j} > 0$ tienden a:

$$p(c_n \neq c_{n'} \; \forall n' \neq n | c_{n'}, \alpha) = \frac{\alpha}{N - 1 + \alpha}. \tag{4.76}$$

Esto muestra que la distribución de la clase a priori para los componentes que tienen asociados observaciones diferentes a y_n es proporcional al número de tales observaciones, mientras que la distribución a priori para el resto de las clases depende sólo de α y del número de observaciones, N. Es importante notar que la tratabilidad analítica de la integral en (4.69) es esencial, dado que nos permite trabajar directamente con un número finito de variables indicadoras en vez de utilizar el número infinito de proporciones de la mezcla [52]. De igual forma, es posible combinar la función de verosimilitud del modelo (ec. 4.50) condicionada en los indicadores con la distribución a priori de las clases (ec. 4.75 ó ec. 4.76, de acuerdo al caso) para obtener la distribución posterior de los indicadores. Para el caso en que $h_{n',j} > 0$, tal distribución posterior toma la forma:

$$\begin{aligned} p(c_n = j | c_{n'}, \mu_j, s_j, \alpha) &\propto p(c_n = j | c_{n'}, \alpha) p(y_n | \mu_j, s_j, c_{n'}) \\ &\propto \frac{h_{n',j}}{N - 1 + \alpha} s_j^{1/2} \exp\left(\frac{-s_j (y_n - \mu_j)^2}{2} \right), \end{aligned} \tag{4.77}$$

mientras que para las clases en que no se satisface $h_{n',j} > 0$, la distribución posterior corresponde con:

$$\begin{aligned} p(c_n \neq c_{n'} \forall n' \neq n | c_{n'}, \lambda, r, \beta, w, \alpha) &\propto p(c_n \neq c_{n'} \forall n' \neq n | c_{n'}, \alpha) \\ &\times \int p(y_n | \mu_j, s_j) p(\mu_j, s_j | \lambda, r, \beta, w) d\mu_j ds_j. \end{aligned} \tag{4.78}$$

Ahora bien, la función de verosimilitud para el componente j cuando $h_{n',j} > 0$ sigue una distribución Gaussiana con componentes μ_j y s_j, mientras que la función de

verosimilitud concerniente a las clases no-representadas (i.e., aquellas que no tienen parámetros asociados) se obtiene por integración sobre su distribución a priori (ecns. 4.52 y 4.53). Es importante notar que no es necesario diferenciar entre las infinitas clases no-representadas dado que la distribución de sus parámetros es idéntica. Sin embargo, esta integral es intratable. Neal [43] sugiere muestrear de tales distribuciones a priori para generar un estimado de la probabilidad de generar una nueva clase. Rasmussen [52] afirma que este procedimiento efectivamente genera parámetros para las clases no-representadas. Dado que este estimador Monte Carlo es insesgado, la cadena resultante muestreará exactamente de la distribución deseada, sin importar el número de muestras que se utilicen para aproximar la integral.

Existen tres posibles situaciones al evaluar las distribuciones posteriores de las clases, dependiendo del número de observaciones asociadas a la clase, como se muestra a continuación:

- si $h_{n',j} > 0$, entonces hay observaciones asociadas a la clase j, por lo que la distribución posterior de la clase se obtiene mediante (4.77).
- si $h_{n',j} = 0$ y $c_n = j$, entonces y_n es la única observación asociada a la clase j; dado que no hay otros parámetros asociados a la clase, se debe tratar como una clase no representada, pero utilizando sus parámetros en vez de muestrearlos.
- si $h_{n',j} = 0$ y $c_n \neq j$, entonces la clase no está representada, de modo que los valores para los componentes son muestreados de las distribuciones a priori de los parámetros (ecns. 4.51 y 4.58).

De esta forma, todas las clases tienen parámetros asociados, por lo que es posible evaluar tanto sus funciones de verosimilitud como sus distribuciones a priori. Las funciones de verosimilitud de las clases siguen una distribución Gaussiana:

$$p(y_n|\mu_j, s_j) = \mathcal{N}(\mu_j, s_j^{-1}). \tag{4.79}$$

Finalmente, las distribuciones a priori para las clases en que $h_{n',j} > 0$ se determinan mediante (4.80), y mediante (4.81) para el resto de las clases.

$$p(y_n|\alpha) = \frac{h_{i',j}}{(n-1+\alpha)} \tag{4.80}$$

$$p(y_n|\alpha) = \frac{\alpha}{(n-1+\alpha)} \tag{4.81}$$

Una nueva clase es introducida en el modelo cuando se seleccione una clase no representada, mientras que las clases se remueven cuando quedan vacías.

4.3.3 Generalización multivariada del método

Hasta el momento se han derivado las distribuciones posteriores bajo la suposición de que las observaciones disponibles son univariadas. Sin embargo, la generalización del método a observaciones multivariadas es directa. Las medias, μ_j, y las precisiones, s_j, de la mezcla se vuelven vectores y matrices, respectivamente, mientras que sus distribuciones a priori y sus distribuciones posteriores se vuelven Gaussianas multivariadas y Wishart. De la misma forma, los hiperparámetros de las medias, λ y r, se vuelven vectores y matrices, y sus distribuciones se vuelven Gaussianas multivariadas y Wishart, respectivamente. Por su parte, los hiperparámetros w de las precisiones se vuelven matrices con distribuciones Wishart. El hiperparámetro β de las precisiones continúa siendo escalar, con la distribución a priori en $(\beta - d + 1)^{-1}$, siendo gamma con media $1/d$. La dimensionalidad de las observaciones está representada por d. De esta forma, para d dimensiones se tiene que:

$$(\beta + d - 1)^{-1} \sim \mathcal{G}(1, 1). \tag{4.82}$$

Si se define una variable g, tal que $g = \beta - d + 1$, entonces:

$$p(g) \propto g^{-3/2}, \tag{4.83}$$

y la distribución posterior de g es:

$$\begin{aligned} p(g|\boldsymbol{S}_1,\dots\boldsymbol{S}_k,\boldsymbol{w}) \;\propto\; & (g+d-1)^{-3/2}\;|\boldsymbol{w}|^{(g+d-1)k/2} \\ & \times\exp\left[\;\left(\frac{d}{2(g+d-1)}\right)\left(\frac{g+d-1}{2}\right)^{(g+d-1)kd/2}\;\right] \\ & \times\prod_{j=1}^{k}\frac{|\boldsymbol{S}_j|^{\frac{g}{2}-1}\exp\left(-\dfrac{(g+d-1)\;\mathrm{tr}(\boldsymbol{W}\boldsymbol{S}_j)}{2}\right)}{\displaystyle\prod_{i=0}^{d-1}\Gamma\left(\frac{g+i}{2}\right)} \end{aligned} \tag{4.84}$$

Por tanto, la función logarítmica de la distribución posterior de g puede evaluarse mediante:

$$\begin{aligned} \ln p(g|\boldsymbol{S}_1,\dots\boldsymbol{S}_k,\boldsymbol{W}) \;\propto\; & -\frac{3}{2}\ln(g+d-1)\;\frac{(g+d-1)k}{2}\ln|\boldsymbol{W}| \\ & +\left(\frac{d}{2(g+d-1)}\right)\left(\frac{g+d-1}{2}\right)^{(g+d-1)kd/2} \\ & +\sum_{j=1}^{k}\frac{|\boldsymbol{S}_j|^{\frac{g}{2}-1}\exp\left(-\dfrac{(g+d-1)\;\mathrm{tr}(\boldsymbol{W}\boldsymbol{S}_j)}{2}\right)}{\displaystyle\prod_{i=0}^{d-1}\Gamma\left(\frac{g+i}{2}\right)} \end{aligned} \tag{4.85}$$

Para generar muestras para β basta con muestrear g mediante un muestreo de Gibbs con paso Metrópolis y recordar que $\beta = g + d - 1$. En la Sección 5.3 se presenta el mecanismo para generar muestras para β para el caso multivariado. El resto de las variables permanecen sin cambio.

No obstante, la generalización multivariada del método puede ser complicada de implementar, especialmente por (i) la aparición de falta de simetría numérica en las matrices, (ii) conflictos en la precisión computacional conforme diferentes matrices se aproximan a la no-singularidad (sin llegar a ser no-singulares) y (iii) algunas restricciones en las distribuciones multivariadas, como lo es el caso de la distribución Wishart. En el Apéndice A se presenta una descripción de nuestra implementación

computacional de la versión multivariada de este método, con observaciones y sugerencias que incorporamos para resolver posibles problemas como los recién descritos. Este apéndice sirve como una guía para la implementación del método. Adicionalmente, en el apéndice se detallan las distribuciones de este capítulo generalizadas al caso multivariado, y se hace referencia una referencia cruzada con las distribuciones univariadas en este capítulo para que la generalización sea más sencilla de visualizar.

4.3.4 Mezcla infinita de Gaussianas como modelo de predicción

Para muestrear de las distribuciones posteriores se aplica el muestreo de Gibbs (Sección 2.2.1). Así, el modelo de mezclas infinitas de Gaussianas se inicializa con un único componente y se computan un número determinado de ciclos de Gibbs, en donde los parámetros e hiperparámetros son actualizados muestreando a partir de sus distribuciones posteriores y el número de componentes del modelo varía. Cuando el muestreo de Gibbs termina, se tiene un conjunto de muestras provenientes del modelo de mezclas infinitas que se asume genera la muestra observada, $\mathcal{D}$. Como se mencionó en la Sección 2.2.1, al aplicar técnicas MCMC es necesario eliminar las muestras que provienen de la cadena de Markov antes que ésta converja, etapa denominada en literatura como *burn-in*. El número de componentes del modelo, k, proporciona una idea acerca de cuándo se alcanza el estado estacionario. Una vez que se han descartado las muestras pertenecientes al estado transiente y que se ha determinado que la cadena ha alcanzado la estacionalidad, es posible determinar el número de componentes más probable, $\langle k \rangle$, que corresponde con aquel que ocurre con una mayor frecuencia en el conjunto remanente de muestras. De este conjunto, sólo aquellas que tengan $\langle k \rangle$ componentes serán finalmente utilizadas para realizar inferencia, de modo que la cantidad de muestras que se necesitan generar por este método varían de acuerdo al problema que se desea resolver. Las distribuciones de probabilidad necesarias para labores de inferencia pueden ser derivadas de la distribución conjunta descrita en (4.50). Para utilizar el modelo de mezclas infini-

tas de Gaussianas como modelo de predicción, después de la etapa de aprendizaje, es necesario introducir finalmente la distinción entre observaciones (o atributos) y objetivos.

El vector de medias $\boldsymbol{\mu}_j$ puede reescribirse como $\boldsymbol{\mu}_j = (\boldsymbol{\mu}_j^x, \boldsymbol{\mu}_j^t)^T$, donde $\boldsymbol{\mu}_j^x$ es el vector que contiene las medias de las observaciones del componente j, tal que $\boldsymbol{\mu}_j^x = (\boldsymbol{\mu}_j^{x_1}, \ldots, \boldsymbol{\mu}_j^{x_q})^T$ y $\boldsymbol{\mu}_j^t$ es el vector que contiene las medias de los objetivos, tal que $\boldsymbol{\mu}_j^t = (\boldsymbol{\mu}_j^{t_1}, \ldots, \boldsymbol{\mu}_j^{t_u})^T$. De manera similar, la matriz de precisión de la clase j puede reescribirse mediante:

$$\boldsymbol{S}_j = \begin{bmatrix} \boldsymbol{S}_j^{xx} & (\boldsymbol{S}_j^{tx})^T \\ \boldsymbol{S}_j^{tx} & \boldsymbol{S}_j^{tt} \end{bmatrix} \qquad \forall j, \tag{4.86}$$

donde, a su vez, $\boldsymbol{S}_j^{xx}$ simboliza la submatriz que contiene las precisiones de las observaciones, $\boldsymbol{S}_j^{tt}$ la submatriz conformada por las precisiones de los objetivos y $\boldsymbol{S}_j^{tx}$ la submatriz que involucra las precisiones entre observaciones y objetivos. Es decir:

$$\boldsymbol{S}_j^{xx} = \begin{bmatrix} \boldsymbol{S}_j^{x_1x_1} & \ldots & (\boldsymbol{S}_j^{x_qx_1})^T \\ \vdots & \ddots & \vdots \\ \boldsymbol{S}_j^{x_1x_q} & \ldots & \boldsymbol{S}_j^{x_qx_q} \end{bmatrix} \qquad \forall j, \tag{4.87}$$

$$\boldsymbol{S}_j^{tt} = \begin{bmatrix} \boldsymbol{S}_j^{t_1t_1} & \ldots & (\boldsymbol{S}_j^{t_ut_1})^T \\ \vdots & \ddots & \vdots \\ \boldsymbol{S}_j^{t_1t_u} & \ldots & \boldsymbol{S}_j^{t_ut_u} \end{bmatrix} \qquad \forall j, \tag{4.88}$$

$$\boldsymbol{S}_j^{tx} = \begin{bmatrix} \boldsymbol{S}_j^{t_1x_1} & \ldots & (\boldsymbol{S}_j^{t_ux_1})^T \\ \vdots & \ddots & \vdots \\ \boldsymbol{S}_j^{t_1x_q} & \ldots & \boldsymbol{S}_j^{t_ux_q} \end{bmatrix} \qquad \forall j. \tag{4.89}$$

La relación entre la matriz de covarianza y la matriz de precisiones del componente j es simplemente $\boldsymbol{\Sigma}_j = (\boldsymbol{S}_j)^{-1}$, por lo tanto:

$$\boldsymbol{S}_j = \begin{bmatrix} \boldsymbol{S}_j^{xx} & (\boldsymbol{S}_j^{tx})^T \\ \boldsymbol{S}_j^{tx} & \boldsymbol{S}_j^{tt} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_j^{xx} & (\boldsymbol{\Sigma}_j^{tx})^T \\ \boldsymbol{\Sigma}_j^{tx} & \boldsymbol{\Sigma}_j^{tt} \end{bmatrix}^{-1} \qquad \forall j. \tag{4.90}$$

La distinción entre observaciones y objetivos afecta también al conjunto de entrenamiento, de modo que $\{\boldsymbol{X}_N, \boldsymbol{T}_N\} = \mathcal{D}$. En el resto de esta sección se abrevia la distribución conjunta de las observaciones y los objetivos dados los parámetros de los componentes, $p(\boldsymbol{X}_N, \boldsymbol{T}_N|\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, \boldsymbol{S}_1, \ldots, \boldsymbol{S}_k, \pi_1, \ldots, \pi_k)$, mediante la distribución conjunta $p(\boldsymbol{X}_N, \boldsymbol{T}_N)$. De acuerdo al modelo (ec. 4.50), la distribución conjunta de las observaciones y los objetivos para el componente j sigue una distribución Gaussiana:

$$p_j(\boldsymbol{X}_N, \boldsymbol{T}_N) = \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{S}_j^{-1}) \qquad \forall j. \tag{4.91}$$

Aplicando el teorema de Bayes (Teorema 2.1), la distribución de los objetivos dadas las observaciones para el componente j en la mezcla es:

$$p_j(\boldsymbol{T}_N|\boldsymbol{X}_N) = \frac{p_j(\boldsymbol{T}_N, \boldsymbol{X}_N)}{p_j(\boldsymbol{X}_N)} \qquad \forall j, \tag{4.92}$$

la cual tiene distribución Gaussiana, al seguir $p(\boldsymbol{X}_N, \boldsymbol{T}_N)$ una distribución Gaussiana. Esta distribución corresponde con:

$$p_j(\boldsymbol{T}_N|\boldsymbol{X}_N) = \mathcal{N}\left(\boldsymbol{\mu}_j^{t|x}, (\boldsymbol{S}_j^{tt})^{-1}\right) \qquad \forall j, \tag{4.93}$$

en donde la media $\boldsymbol{\mu}_j^{t|x}$ de esta distribución Gaussiana se evalúa mediante:

$$\boldsymbol{\mu}_j^{t|x} = \boldsymbol{\mu}_j^t - (\boldsymbol{S}_j^{tt})^{-1}\boldsymbol{S}_j^{tx}\ (\boldsymbol{X}_N - \boldsymbol{\mu}_j^x) \qquad \forall j. \tag{4.94}$$

Extendiendo estos cálculos a los k componentes, la distribución condicional de los objetivos dadas las observaciones es:

$$p(\boldsymbol{T}_N|\boldsymbol{X}_N) = \sum_{j=1}^{k} \pi'_j \; p_j(\boldsymbol{T}_N|\boldsymbol{X}_N), \tag{4.95}$$

en donde a su vez:

$$\pi'_j = \frac{\pi_k \; \mathcal{N}(\boldsymbol{X}_N|\boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})}{\sum\limits_{j=1}^{k} \pi_k \; \mathcal{N}(\boldsymbol{X}_N|\boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})}. \tag{4.96}$$

Sea $\boldsymbol{x}^f$ una observación para la cual se desean inferir los objetivos $\boldsymbol{t}^f$. La salida del modelo, y por tanto la inferencia sobre los objetivos $\boldsymbol{t}^f$, corresponde con el valor esperado de la distribución condicional de los objetivos dada la observación $\boldsymbol{x}^f$:

$$\langle \boldsymbol{t}^f|\boldsymbol{x}^f \rangle = \sum_{j=1}^{k} \pi_j^f \; \boldsymbol{\mu}_j^f \qquad \forall f, \tag{4.97}$$

en donde:

$$\boldsymbol{\mu}_j^f = \boldsymbol{\mu}_j^t - (\boldsymbol{S}_j^{tt})^{-1} \boldsymbol{S}_j^{tx} \; (\boldsymbol{x}_f - \boldsymbol{\mu}_j^x) \qquad \forall f, \tag{4.98}$$

$$\pi_j^f = \frac{\pi_k \; \mathcal{N}(\boldsymbol{x}_f|\boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})}{\sum\limits_{j=1}^{k} \pi_k \; \mathcal{N}(\boldsymbol{x}_f|\boldsymbol{\mu}_j^x, \boldsymbol{\Sigma}_j^{xx})} \qquad \forall f. \tag{4.99}$$

4.4 Bootstrap

El *bootstrap* es un método de remuestreo (paramétrico o no-paramétrico) que aproxima la distribución muestral de un estadístico. Es, en esencia, una implementación computacional de máxima verosimilitud [24] que ha ganado popularidad y una gran aceptación en el área de la estadística aplicada durante los últimos años debido principalmente a que permite evaluar estimadores de máxima verosimilitud

y otras cantidades de interés en situaciones en que no hay otros métodos aplicables, o cuando las aproximaciones clásicas no proporcionan resultados satisfactorios [16]. El *bootstrap* es una herramienta estadística computacionalmente intensiva que puede construir estimadores que serían difícil de obtener por otros medios.

En la estadística clásica frecuentista se hacen suposiciones acerca de la estructura de la población (e.g., la suposición de normalidad en una población), a partir de las cuales se deriva la distribución probabilística del estimador para algún parámetro poblacional, θ. Este enfoque es intratable en un gran número de circunstancias, por lo que usualmente se recurre a derivar su distribución asintótica. No obstante, esta metodología tiene algunas deficiencias. Si la suposición de la estructura de la población es violada, entonces la distribución desarrollada para el estimador es inapropiada, mientras que la aplicación de la distribución asintótica puede no ser lo suficientemente precisa para muestras pequeñas. En contraste, el *bootstrap* es una técnica que permite estimar experimentalmente la distribución muestral de un estadístico, sin la necesidad de hacer suposiciones acerca de la forma de la población y sin tener que derivar esta distribución muestral explícitamente, lo que representa una gran ventaja en comparación con otros métodos cuando se tienen observaciones que no se comportan normalmente, cuando sus distribuciones muestrales son difíciles de derivar o cuando se tienen muestras de tamaño pequeño.

Para comprender el *bootstrap* supóngase por un momento que es posible tomar una cantidad muy grande de muestras de la población de interés, de tal forma que se podría tener un buen estimado de la distribución muestral de un estadístico en particular. Este caso es infactible en muchos escenarios, donde tomar una muestra es un proceso costoso o destructivo. La idea detrás del *bootstrap* es utilizar la información de una sola muestra como un reemplazo de la población, es decir, tomar muestras subsecuentes con reemplazo a partir de la observación original. Es necesario que el muestreo sea con reemplazo, ya que de otra forma se estaría reproduciendo la misma observación. Sea $\boldsymbol{s}$ una muestra tomada de la población $\boldsymbol{P}$, tal que $\boldsymbol{s} = \{x_1, \ldots, x_N\}$, y sea $\hat{\theta} = f(\boldsymbol{s})$ el estimador de algún parámetro poblacional $\theta = f(\boldsymbol{P})$. Si se toma una

muestra de tamaño N_B con reemplazo del conjunto $\boldsymbol{s}$, tal que $\boldsymbol{s}_1^* = \{x_{1,1}^*, \ldots, x_{1,N_B}^*\}$, entonces cada elemento x_n tiene una probabilidad $1/N$ de ser seleccionado, imitando la selección original de la muestra tomada de la población. Este procedimiento de remuestreo se repite R veces, siendo R un número lo suficientemente grande que permita generar una gran cantidad de muestras *bootstrap*. Posteriormente se computa el estimador $\hat{\theta}$ para cada una de ellas. Sea $\boldsymbol{s}_j^*$ la j-ésima muestra *bootstrap*, tal que $\boldsymbol{s}_j^* = \{x_{j,1}^*, \ldots, x_{j,N}^*\}$, el j-ésimo estimador corresponde entonces con $\hat{\theta}_j^* = f(\boldsymbol{s}_j^*)$. La distribución de $\hat{\theta}_j^*$ alrededor del estimador original $\hat{\theta}$ es análoga a la distribución del estimador $\hat{\theta}$ alrededor del parámetro θ. Por ejemplo, el valor medio del estadístico *bootstrap* es:

$$\langle \hat{\theta}^* \rangle = E^*(\hat{\theta}^*) = \frac{1}{R} \sum_{j=1}^{R} \hat{\theta}_j^*, \tag{4.100}$$

el cual estima la esperanza del estadístico *bootstrap*, de modo que $\hat{D}^* = \langle \hat{\theta}^* \rangle - \hat{\theta}$ es un estimado del sesgo de $\hat{\theta}$, es decir, $\hat{\theta} - \theta$. De manera similar, la varianza *bootstrap*:

$$\bar{V}^*(\hat{\theta}^*) = \frac{1}{R-1} \sum_{j=1}^{R} (\hat{\theta}_j^* - \langle \hat{\theta} \rangle^*)^2, \tag{4.101}$$

estima la varianza muestral de $\hat{\theta}$. De modo que para hacer inferencia mediante una técnica *bootstrap* es necesario determinar la función $f(\cdot)$ para el estimador del parámetro, $\hat{\theta} = f(\boldsymbol{s})$, es decir, se necesita determinar el modelo.

Existen dos fuentes de error en la inferencia mediante *bootstrap*: (i) el error inducido por utilizar una muestra particular $\boldsymbol{s}$ para representar la población, y (ii) el error muestral producido por no analizar todas las posibles muestras *bootstrap*, el cual puede ser controlado tomando una gran cantidad de muestras. Para una explicación más detallada acerca de las técnicas *bootstrap* véanse [18, 16, 32, 65]. En la Sección 5.3 se detalla el uso del método *bootstrap* utilizado en esta tesis.

Capítulo 5

Evaluación Computacional

El objetivo de esta tesis es estudiar el comportamiento de la capacidad de pronóstico para diferentes métodos Bayesianos no-lineales en contraste con una técnica popular de remuestreo, el método *bootstrap*, así como la forma en que esta capacidad es afectada por el tamaño del conjunto de entrenamiento, donde el efecto del ruido y la no-linealidad inherentes a los problemas de estudio tiende a hacerse más visible conforme se disminuye el número de observaciones. Para esta labor, se seleccionaron tres problemas retadores provenientes de diferentes áreas del conocimiento: el XOR continuo, la afinidad en acoplamientos enzimáticos y la concentración de metales pesados en la capa superficial del suelo. El ambiente computacional en que se implementaron los métodos Bayesianos se presenta en la Sección 5.1, mientras que en la Sección 5.2 se presenta una descripción de las observaciones empleadas en esta tesis como conjuntos de entrenamiento. Posteriormente, en la Sección 5.3 se proporciona una descripción de los aspectos técnicos de nuestra implementación de los métodos de solución. Finalmente, en la Sección 5.4 se presenta la evaluación computacional y los resultados de este estudio.

5.1 Ambiente de programación y librerías

Los tres métodos Bayesianos descritos en el Capítulo 4 fueron implementados computacionalmente en el lenguaje *R*, que forma parte del proyecto *R Statistical Software* [50]. En el Apéndice A se presenta una descripción concisa de la implemen-

tación realizada, la cual sirve como guía para implementar los elementos necesarios de estos métodos partiendo desde cero. Esto es especialmente útil para el modelo de mezcla infinita de Gaussianas, el cual no puede encontrarse en ninguna librería y cuya implementación computacional multivariada es difícil, siendo una de las contribuciones de esta tesis. Para la evaluación computacional de los problemas de estudio mediante el método *bootstrap* se utilizó la librería *boot* [11, 15] del mismo ambiente de programación, mientras que la rutina de optimización del recocido simulado en los procesos Gaussianos se hizo por medio del método *SANN* de la función *optim*, perteneciente a la librería *stats* [50]. Los experimentos fueron realizados en *R 2.13.0* mediante un servidor con cuatro procesadores Intel® Xeon® E5320 a 1.86 GHz, con 4 Gb de memoria RAM DDR2.

5.2 Descripción de los conjuntos de entrenamiento

En esta sección se proporciona información acerca de las observaciones empleadas en esta tesis como conjuntos de entrenamiento para los tres problemas de estudio propuestos. Se indica además la construcción de subconjuntos que permiten analizar la dependencia de la capacidad de pronóstico con el tamaño del conjunto muestral.

XOR continuo. Para el problema XOR continuo, el conjunto de entrenamiento consistió en la generación aleatoria de $N = 150$ pares ordenados, $\{x_n^1, x_n^2\}_{n=1}^N$, con distribución uniforme en el intervalo $[0.1, 1]$. Se decidió generar un total de 150 observaciones debido a que es la cantidad que se utiliza en experimentos previos para entrenar una red neuronal [59] y que ésta sea capaz de aproximar adecuadamente la función de discriminación apropiada [59]. Para cada par ordenado en N se evaluó la salida de la compuerta lógica digital ($y_n = \{0, 1\}$) mediante la función de discriminación óptima (ec. 3.1), la cual se deriva de las reglas lógicas que separan los pares

ordenados. De esta forma, la función general que describe este problema es:

$$y = f(x^1, x^2), \tag{5.1}$$

donde $f(x^1, x^2)$ corresponde a su vez con la parametrización que hace cada uno de los métodos de solución propuestos. Por ejemplo, para el caso más sencillo, si se desea aproximar 5.1 mediante una función lineal, entonces:

$$y = f_{\text{lineal}}(x^1, x^2) = c_0 + c_1 x^1 + c_2 x^2. \tag{5.2}$$

Afinidad de acoplamientos enzimáticos. Por su parte, el conjunto de entrenamiento para el problema de estimación de la afinidad de enzimas y sustratos fue tomado de [36], y contiene la energía de interacción como indicador de la afinidad para 27 enzimas y 119 sustratos, seleccionados de entre los metabolitos de la bacteria *Escherichia coli*. Una función elemental que describe la energía de interacción (e) para el complejo formado por la enzima p y el sustrato s es de la forma:

$$e = f(p, s), \tag{5.3}$$

en donde $f(p, s)$ es la función que desea parametrizarse mediante los métodos de solución para aproximar la afinidad de un complejo proteico.

Concentración de metales pesados en la capa superficial del suelo. Finalmente, el conjunto de entrenamiento para el problema de pronóstico de la concentración de metales pesados en el suelo contiene los siguientes atributos: el uso que se le da al terreno (us), el tipo de roca superficial encontrado (rc), y la concentración de cadmio (cd), cobalto (co), cromo (cr), níquel (ni), plomo (pb) y zinc (zn) en 359 ubicaciones distintas. Estas observaciones fueron tomadas de [22]. El problema consiste en aproximar un par de funciones que estimen la concentración de cadmio mediante la concentración de níquel y zinc, y la concentración de cobre mediante la

concentración de níquel, zinc y plomo, además del uso del terreno y el tipo de piedra. Es decir, mediante los métodos de solución propuestos se desea aproximar f_1 y f_2 en 5.4 y 5.5 para determinar, respectivamente, la concentración de cadmio y cobre.

$$cd = f_1(us, rc, ni, zn), \tag{5.4}$$

$$cu = f_2(us, rc, ni, zn, pb). \tag{5.5}$$

Construcción de subconjuntos de entrenamiento. En esta tesis se utiliza un porcentaje variable en la cantidad de observaciones que forman parte del conjunto de entrenamiento para analizar la dependencia de cada método con respecto al tamaño del conjunto muestral. Las observaciones restantes fueron utilizadas como un conjunto de prueba para evaluar la capacidad de pronóstico de cada modelo entrenado. Específicamente, se construyeron (de manera aleatoria) los conjuntos de prueba y entrenamiento mostrados en la Tabla 5.1.

Nomenclatura	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
Entrenamiento (%)	10	20	30	40	50	60	70	80	90
Prueba (%)	90	80	70	60	50	40	30	20	10

Tabla 5.1: Nomenclatura de los subconjuntos de entrenamiento. Entrenamiento y Prueba indican el porcentaje de observaciones que forman parte de los conjuntos de entrenamiento y de prueba, respectivamente.

5.3 Aspectos técnicos de la implementación de los métodos de solución

En esta sección se proporciona información técnica acerca de nuestra implementación de los métodos Bayesianos, como son el número de ciclos de Gibbs cuando

es necesario un muestreo, la cantidad de iteraciones máxima cuando se aplica una rutina de optimización, etcétera.

Redes neuronales artificiales (ANN). Para la aplicación de las ANN como método se solución se utilizó una arquitectura con una capa oculta, la cual contenía 20 neuronas ocultas independientemente del problema bajo estudio. Esta arquitectura ha proporcionado buenos resultados en experimentos previos, y evidencia las observaciones de Neal en nuestros resultados. Como se describió en la Sección 4.1.4, para computar la salida de una ANN se utiliza una aproximación mediante muestras generadas por un muestreo de Gibbs con paso Metrópolis. En nuestra implementación, se generó una cadena de Markov realizando 500 ciclos de Gibbs, de los cuales se eliminaron alrededor de las 200 primeras por pertenecer al estado transitorio. El parámetro de escala del algoritmo se ajustó de tal forma que se aceptaran aproximadamente la mitad de los candidatos.

Procesos Gaussianos (GP). En la Sección 4.2.5 se presentó la alternativa de maximizar la verosimilitud logarítmica marginal como una técnica útil para aproximar la distribución condicional de interés cuando no se tiene información a priori acerca de los parámetros, mediante la aplicación de los parámetros más probables. Para llevar a cabo el problema de optimización se aplica el recocido simulado, por tener ventajas notorias que han sido discutidas en la Sección 4.2.5. Para la técnica del recocido simulado se permitió que la función *optim* seleccionara la temperatura inicial de acuerdo a sus propios métodos de estimación, y se le permitió una cantidad máxima de 3,000 iteraciones en su búsqueda del conjunto óptimo de parámetros.

Mezcla infinita de Gaussianas (IGMM). Para muestrear de las distribuciones posteriores de α y β, las cuales no son de forma estándar, se aplica nuevamente un muestreo de Gibbs con paso Metrópolis. El objetivo del muestreo de Gibbs en esta ocasión es generar un pequeño conjunto de muestras. En el caso de α se necesita una sola muestra sin ningún tipo de restricciones, de modo que el muestreo termina

cuando la primera perturbación produzca un candidato que sea aceptado. Por otro lado, en el caso de β se muestrea primero la variable g mediante la distribución en (4.84). Dado que β es el parámetro de forma de una distribución Wishart, las muestras de β deben cumplir con un requerimiento característico de tal distribución:

$$\beta \geq d, \tag{5.6}$$

donde d es la dimensionalidad de las observaciones, de modo que $g \geq 1$. Así, en esta tesis se opta por generar una cantidad relativamente pequeña de muestras de g mediante el muestreo de Gibbs con paso Metrópolis, remover aquellos que no satisfacen $g \geq 1$, y finalmente seleccionar de manera aleatoria uno de los valores restantes. Esta metodología ha mostrado un buen desempeño en la práctica. La muestra de β se recupera mediante:

$$\beta = g + d - 1. \tag{5.7}$$

De nueva cuenta, los parámetros de escala del algoritmo se ajustaron de tal forma que fueron aceptados alrededor de la mitad de los candidatos. Por otra parte, el número de ciclos de Gibbs que desempeñó el algoritmo varió de acuerdo al problema y a la sensibilidad que tienen los valores iniciales en la cadena de Markov construida. Para el problema XOR continuo se generaron 8,000 muestras en promedio, descartándose una media de 1,000 ciclos; mientras que para el pronóstico de la afinidad en complejos proteicos se generaron en promedio 15,000 muestras, descartándose una media de 5,000 ciclos; finalmente, para la estimación de la concentración superficial de metales pesados se tomaron alrededor de 12,000 muestras, descartándose las primeras 2,000 en promedio. El número de ciclos de Gibbs se determinó de tal forma que se tuvieran un mínimo de 7,000 muestras provenientes de la distribución posterior para el pronóstico, determinando el inicio del estado estable de acuerdo a las pruebas presentadas en la Sección 2.2.

Bootstrap (BTSR). Para la resolución de los problemas mediante BTSR se seleccionó una función de regresión polinómica de la forma:

$$\hat{y} = b_0 + \sum_{i=1}^{v} b_i x^i + \sum_{i=1}^{v} \sum_{j=1}^{v} b_{ij} x^i x^j, \tag{5.8}$$

donde v representa la dimensionalidad de los atributos de entrada. Se permitió que la librería *boot* utilizara sus parámetros por defecto, además de indicarle que realizara 5,000 replicaciones *bootstrap* de la muestra original. La estimación mediante BTSR en esta tesis consiste en ajustar los coeficientes de regresión en (5.8) para cada una de las 5,000 observaciones, para posteriormente computar el M-estimador de Huber [27] de tales coeficientes. Se estudió la inclusión de términos de mayor orden en (5.8), sin embargo, la variación en la capacidad de pronóstico fue mínima, por lo que estos términos fueron omitidos.

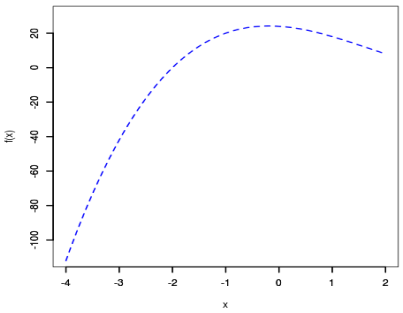
5.4 Evaluación de los métodos de solución

En esta sección se evalúa la capacidad de pronóstico de los métodos de solución al aplicarlos a los problemas de estudio propuestos. Estos problemas de estudio fueron seleccionados de tal forma que la función a aproximar se tratara de una función no-lineal complicada, además de tener características que puedan conducir a resultados intuitivos: las observaciones asociadas al problema XOR continuo están libres de ruido, de tal forma que la variación en la capacidad de pronóstico de los métodos es exclusivamente por efecto de la no-linealidad que existe en sus observaciones, mientras que el problema de los acoplamientos enzimáticos contiene niveles altos tanto de ruido como de no-linealidad, lo que permite observar el desempeño de los métodos Bayesianos para evitar el entrenamiento de un modelo con sobreajuste, y cómo este desempeño se ve afectado conforme se disminuye el tamaño del conjunto de entrenamiento, en comparación con un modelo que tiene la capacidad de evitar el sobreajuste. Finalmente, el problema de concentración de metales pesados contiene atributos de salida correlacionados, lo que representa a su vez una desventaja

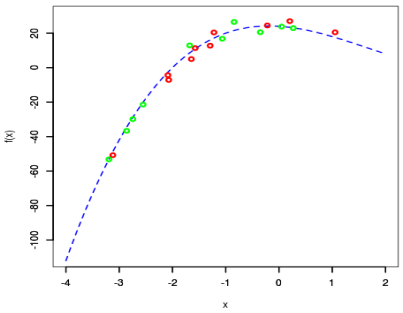
cuando se realizan las tareas de manera individual, siendo un reto principalmente para el desempeño de IGMM. La evaluación experimental de los métodos de solución se muestra en la Figura 5.1. De manera inicial, se selecciona uno de los problemas de estudio presentados en el Capítulo 3 (Figura 5.1(a)), así como un conjunto de entrenamiento y su conjunto de prueba correspondiente de los introducidos en la Tabla 5.1 (Figura 5.1(b)). Posteriormente se selecciona y aplica uno de los métodos de solución detallados en el Capítulo 4 (Figura 5.1(c)). Una vez que se tiene entrenado el modelo que funge como aproximación al problema de estudio (Figura 5.1(d)), se calcula el valor esperado para el conjunto de prueba (Figura 5.1(e)) y se evalúa la capacidad de pronóstico del método bajo estudio mediante el cálculo del NRMSE (Figura 5.1(f)). Este proceso de entrenamiento se repite cinco veces de tal forma que todos los problemas de estudio, los métodos de solución, los conjuntos de entrenamiento y los conjuntos de prueba sean cubiertos. El resto de esta sección presenta los resultados de la evaluación experimental descrita.

5.4.1 XOR continuo

La Tabla 5.2 introduce la capacidad de pronóstico de los métodos de solución para el problema del XOR continuo. La primera columna contiene los distintos métodos seleccionados en esta tesis, mientras que el resto de las columnas (i.e., de la segunda a la décima) representan los conjuntos de entrenamiento construidos tal como se describió en la Sección 5.2. Los elementos de esta tabla indican el NRMSE obtenido por cada método siendo entrenado con cada conjunto de observaciones, y corresponde con el promedio de cinco ejecuciones independientes, variando el conjunto de entrenamiento construido en cada ejecución. La desviación estándar es semejante entre los diferentes métodos, de modo que el promedio se considera representativo. La Figura 5.2 muestra gráficamente estos resultados, de donde pueden extraerse algunas observaciones. Resulta interesante observar que la capacidad de predicción de ANN (en rojo) es muy superior a la de GP (en magenta), situación que no concuerda con las observaciones de Neal. Esto puede atribuirse principalmen-



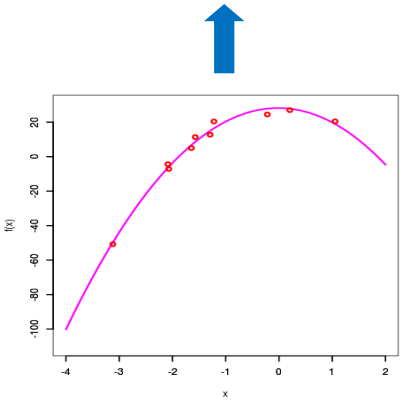
(a) Selección del problema a resolver:
- XOR continuo (Cap. 3.1)
- Complejos enzimáticos (Cap. 3.3)
- Concentración de metales (Cap. 3.2)



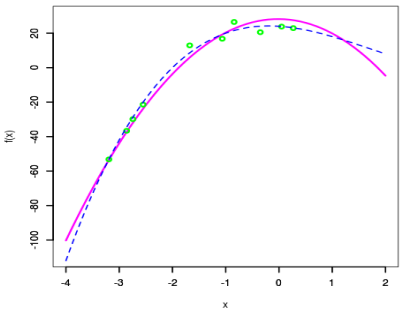
(b) Selección de los conjuntos de entrenamiento (en rojo) y de prueba (en verde), de la Tabla 5.1.

(c) Selección y aplicación del método de solución al conjunto de entrenamiento:
- ANN (Cap. 4.1)
- GP (Cap. 4.2)
- IGMM (Cap. 4.3)
- BTSR (Cap. 4.4)

(d) Determinación de la aproximación (modelo) al problema original.



(e) Cómputo de la salida del modelo para el conjunto de prueba.



(f) Determinación del error cuadrado medio normalizado (NRMSE).

$$NRMSE = \frac{\sqrt{\sum_{i=1}^{n} \frac{(y_{\text{observación}} - y_{\text{modelo}})^2}{n}}}{y^{\max}_{\text{observación}} - y^{\min}_{\text{observación}}}$$

Figura 5.1: Esquema de solución para resolver un problema de estudio. Inicialmente se (a) selecciona el problema a resolver, a partir del cual se (b) extraen los conjuntos de entrenamiento y de prueba. Posteriormente, se (c) selecciona y aplica un método de solución, mediante el cual se (d) realiza una aproximación a la función deseada. Una vez que se cuenta con el modelo entrenado, se (e) determina el valor esperado del modelo sobre el conjunto de prueba y se (f) evalúa la capacidad de generalización computando el NRMSE.

te a que hemos seleccionado la optimización de la verosimilitud logarítmica marginal para entrenar GP, evidenciando la desventaja de utilizar un conjunto óptimo de hiperparámetros en vez de utilizar el valor esperado de su distribución probabilística para este problema en concreto. Por otro lado, el método que muestra la mejor capacidad de pronóstico para todos los conjuntos de observaciones es IGMM (en azul), seguido por las capacidades de pronóstico de ANN y BTSR (en verde). La diferencia promedio en la capacidad de pronóstico para IGMM y ANN es de 0.9, y de 0.8 para IGMM y BTSR, siendo ambas significativas. Para IGMM y ANN puede observarse que parece existir un efecto ocasionado por el tamaño muestral del conjunto de entrenamiento bajo las condiciones de experimentación, al volverse más importantes los efectos del ruido y la no-linealidad en conjuntos pequeños (por debajo de DS4). Para el conjunto de menor tamaño analizado, DS1, IGMM presenta una clara ventaja sobre ANN pero no así sobre BTSR, mientras que para el conjunto de mayor tamaño estudiado, DS9, IGMM muestra una ventaja por encima de ambos métodos. Así, IGMM exhibe una mayor robustez en su capacidad de pronóstico ante el tamaño del conjunto de observaciones, en comparación con el resto de los métodos. En nuestra experiencia, la convergencia de IGMM necesita en ocasiones de un gran número de iteraciones, al tener un mayor efecto los valores iniciales seleccionados, lo que permite explicar el valle que presenta en DS5 y DS6 como una estacionalidad temprana, donde IGMM muestra una capacidad de pronóstico impresionante, muy por encima de los niveles del resto de los métodos. Otra cuestión de interés es que las capacidades de pronóstico de GP y BTSR parecen no resentir el efecto del tamaño del conjunto de entrenamiento.

5.4.2 Afinidad de acoplamientos enzimáticos

La Tabla 5.3 muestra la capacidad de pronóstico al inferir la afinidad de acoplamientos enzimáticos. El formato de esta tabla es semejante al de la tabla anterior. IGMM obtiene la mejor capacidad de pronóstico para todos los conjuntos de entrenamiento, teniendo una ventaja promedio de 0.027, 0.012 y 0.028 sobre ANN, GP

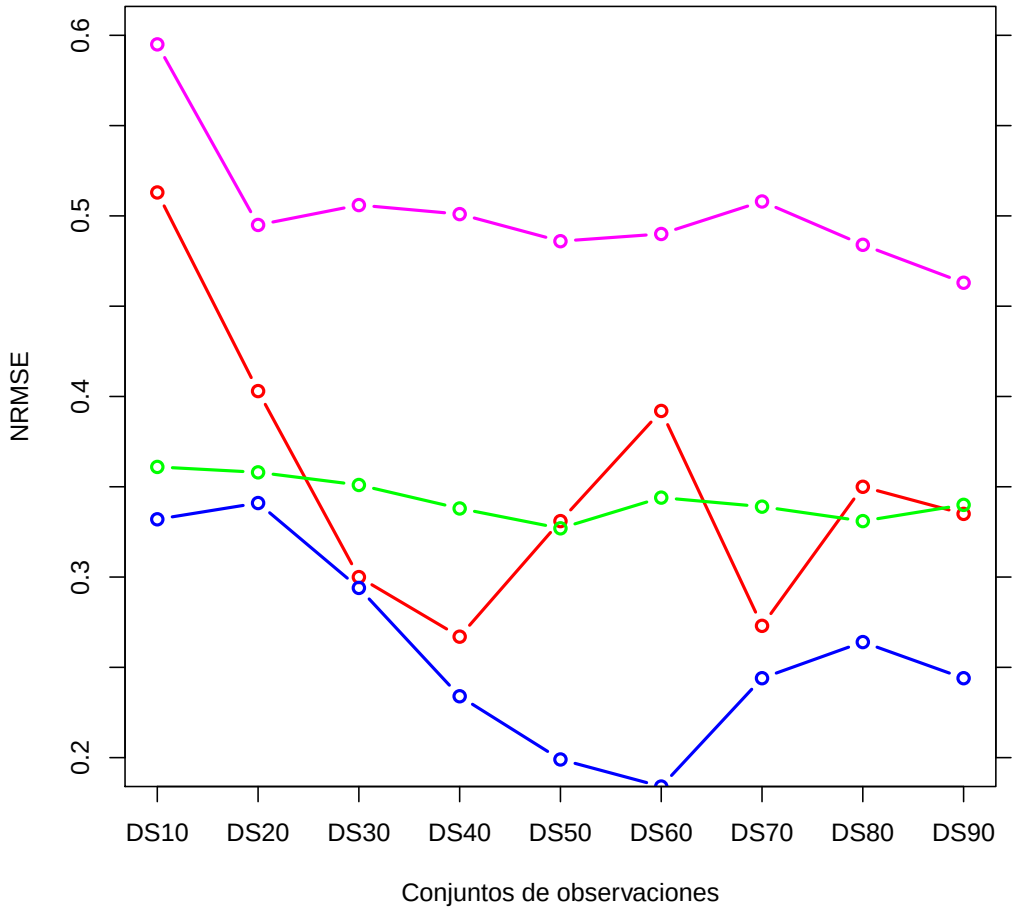


Figura 5.2: NRMSE de los métodos de solución para el problema XOR continuo al variar el tamaño del conjunto de observaciones. En rojo se muestran los resultados de ANN, en magenta los correspondientes a GP, en azul aquellos de IGMM y en verde los pertenecientes a BTSR.

Métodos de	Conjuntos de observaciones								
solución	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
ANN	0.513	0.403	0.300	0.267	0.331	0.392	0.273	0.350	0.335
GP	0.595	0.495	0.506	0.501	0.486	0.490	0.508	0.484	0.463
IGMM	0.332	0.341	0.294	0.234	0.199	0.184	0.244	0.264	0.244
BTSR	0.361	0.358	0.351	0.338	0.327	0.344	0.339	0.331	0.340

Tabla 5.2: NRMSE de los métodos de solución para el problema XOR continuo al variar el tamaño del conjunto de observaciones.

y BTSR, respectivamente. A pesar de presentar la mejor capacidad de pronóstico, la diferencia promedio no es lo suficientemente grande para afirmar que ésta es significativa, como ocurre en el caso del XOR continuo. No obstante, esta diferencia es notoria cuando se tienen conjuntos de observaciones pequeños, por debajo de DS5, exhibiendo IGMM una capacidad de pronóstico considerablemente superior a la de ANN para el conjunto de menor tamaño estudiado, y una ventaja de 0.02 para el resto de los métodos. Por el contrario, para el conjunto de mayor tamaño estudiado, IGMM presenta una mejor capacidad de pronóstico especialmente en comparación con BTSR. Otra cuestión de interés consiste en observar que la capacidad de pronóstico de ANN y GP concuerdan, evidenciando las observaciones de Neal, lo que implica que la rutina de optimización de GP es adecuada para resolver este problema. De manera general, GP muestra una mayor capacidad de pronóstico en comparación con BTSR, mientras que ANN presenta una mejor inferencia que BTSR cuando el tamaño muestral es lo suficientemente grande (por encima de DS5). Nuevamente se observa un valle en los resultados de IGMM para DS7, lo que podría indicar que la capacidad de pronóstico de IGMM puede mejorarse.

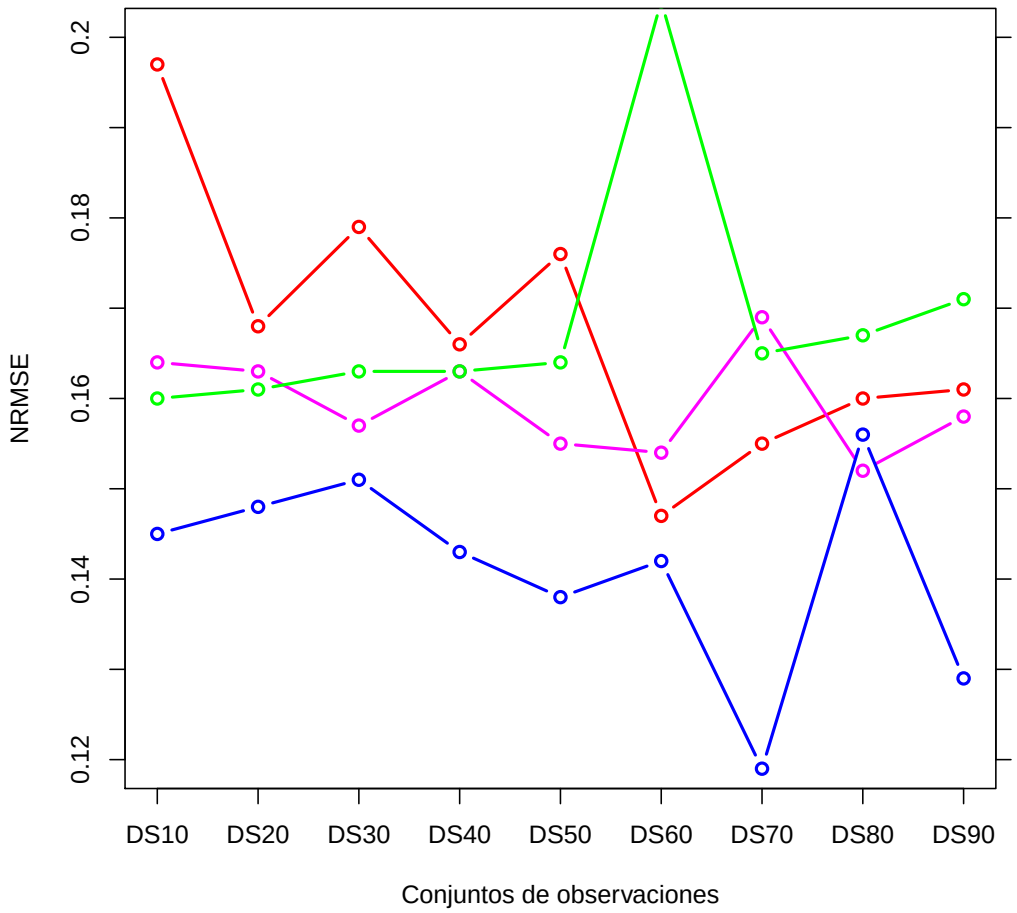


Figura 5.3: NRMSE de los métodos de solución al estimar la afinidad de acoplamiento enzimáticos variando el tamaño del conjunto de observaciones. En rojo se muestran los resultados de ANN, en magenta los correspondientes a GP, en azul aquellos de IGMM y en verde los pertenecientes a BTSR.

Métodos de	Conjuntos de observaciones								
solución	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
ANN	0.197	0.168	0.179	0.166	0.176	0.147	0.155	0.160	0.161
GP	0.164	0.163	0.157	0.163	0.155	0.154	0.169	0.152	0.158
IGMM	0.145	0.148	0.151	0.143	0.138	0.142	0.119	0.156	0.129
BTSR	0.160	0.161	0.163	0.163	0.164	0.204	0.165	0.167	0.171

Tabla 5.3: NRMSE de los métodos de solución al estimar la afinidad de acoplamiento enzimáticos variando el tamaño del conjunto de observaciones.

5.4.3 Concentración de metales pesados en la capa superficial del suelo

Por su parte, las Tablas 5.4 y 5.5 presentan la capacidad de pronóstico al inferir la concentración superficial de cadmio y cobre, respectivamente. El formato de la tabla es semejante a los utilizados anteriormente. Las Figuras 5.4 y 5.5 introducen gráficamente estos resultados. Es importante recordar que ANN, GP y BTSR resuelven cada tarea de manera separada e independiente, mientras que IGMM resuelve ambas tareas simultáneamente. Para la tarea de estimar la concentración de cadmio, cuando el tamaño del conjunto de entrenamiento es pequeño (por debajo de DS3), ANN, IGMM y BTSR tienen una capacidad de pronóstico semejante. Sin embargo, conforme aumenta el tamaño muestral, los resultados de ANN y BTSR divergen de aquellos de IGMM. De este modo, IGMM obtiene nuevamente la mejor capacidad de pronóstico de entre los métodos de solución, mientras que ANN y BTSR muestran capacidades de pronóstico semejantes entre ellos. No obstante, la diferencia promedio entre los errores es de 0.3, una diferencia pequeña en comparación con la correspondiente para el XOR continuo, pero considerablemente mayor a la encontrada en complejos proteicos. De manera similar a lo que ocurre en el XOR continuo, los resultados de ANN y GP divergen notoriamente, evidenciando nuevamente la desventaja de emplear la rutina de optimización para resolver GP.

El efecto del tamaño muestral en la capacidad de pronóstico de IGMM vuelve a hacerse presente. En cuanto a la tarea de estimar la concentración de cobre, cuando el tamaño del conjunto de observaciones es pequeño (por debajo de DS6), IGMM y BTSR muestran la mejor capacidad de pronóstico. Sin embargo, conforme el tamaño muestral aumenta, los resultados de BTSR divergen de aquellos de IGMM. Una vez más, IGMM obtiene la mejor capacidad de pronóstico. Por su parte, ANN obtiene una capacidad de pronóstico relativamente cercana a ambos métodos, difiriendo sus resultados por un sólo orden de magnitud para conjuntos pequeños de observaciones, mientras que al aumentar el número de observaciones sus resultados se aproximan a aquellos de BTSR. De manera semejante a la tarea anterior, los resultados de ANN y GP divergen. Es importante observar que IGMM parece ser robusto ante el tamaño muestral del conjunto de observaciones.

Métodos de	Conjuntos de observaciones								
solución	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
ANN	0.148	0.142	0.131	0.189	0.135	0.132	0.121	0.164	0.193
GP	0.202	0.234	0.230	0.225	0.199	0.217	0.290	0.298	0.271
IGMM	0.142	0.142	0.128	0.139	0.111	0.106	0.121	0.117	0.112
BTSR	0.140	0.138	0.135	0.152	0.145	0.135	0.146	0.147	0.175

Tabla 5.4: NRMSE de los métodos de solución al pronosticar la concentración de cadmio conforme varia el tamaño del conjunto de observaciones.

De manera adicional, se utilizan los conjunto de observaciones empleados en el estudio comparativo presentado en [2] para entrenar y evaluar IGMM, al ser el método que tiene la mejor capacidad de pronóstico en nuestro estudio, de modo que los resultados de IGMM son comparables con aquellos que involucra tal estudio: procesos Gaussianos independientes, aproximaciones condicionales parcialmente independientes [29], procesos Gaussinos completos y *cokriging* ordinario (véase [2] para una descripción detallada). El conjunto de entrenamiento contiene 259 observaciones, mientras que el conjunto de evaluación consiste en 100 muestras. El error medio

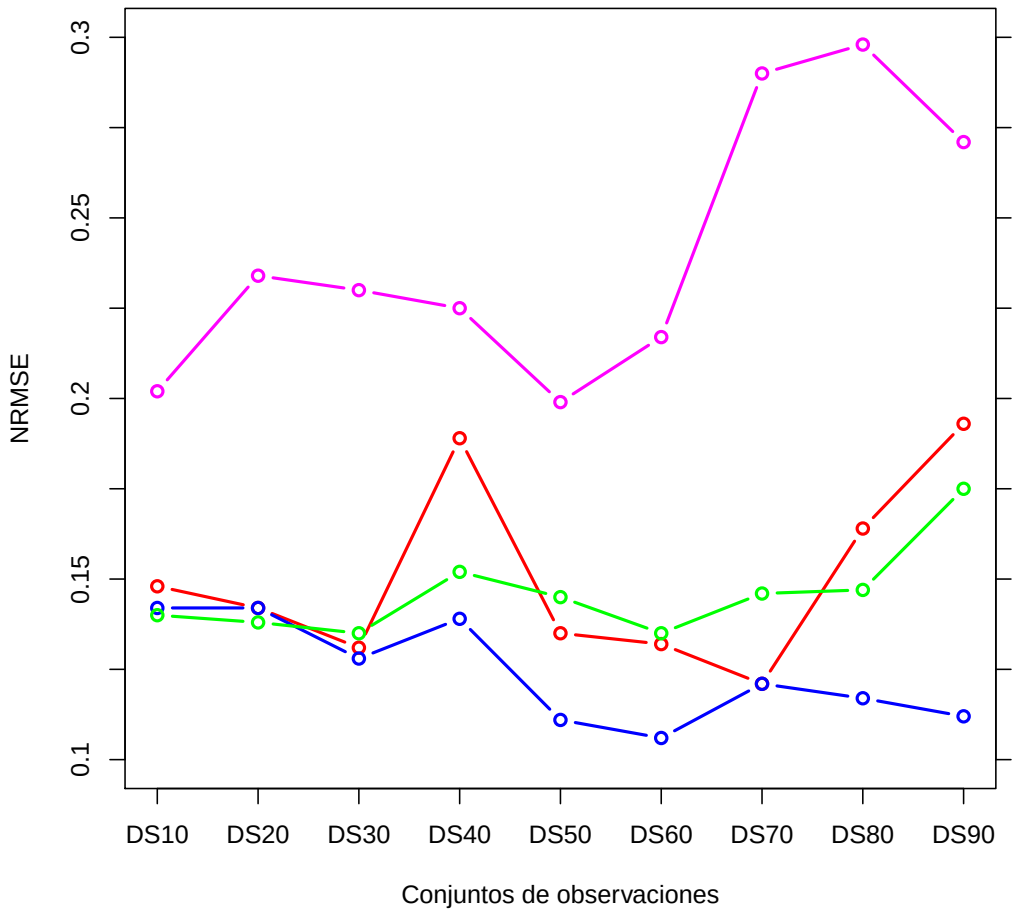


Figura 5.4: NRMSE de los métodos de solución al pronosticar la concentración de cadmio conforme varia el tamaño del conjunto de observaciones. En rojo se muestran los resultados de ANN, en magenta los correspondientes a GP, en azul aquellos de IGMM y en verde los pertenecientes a BTSR.

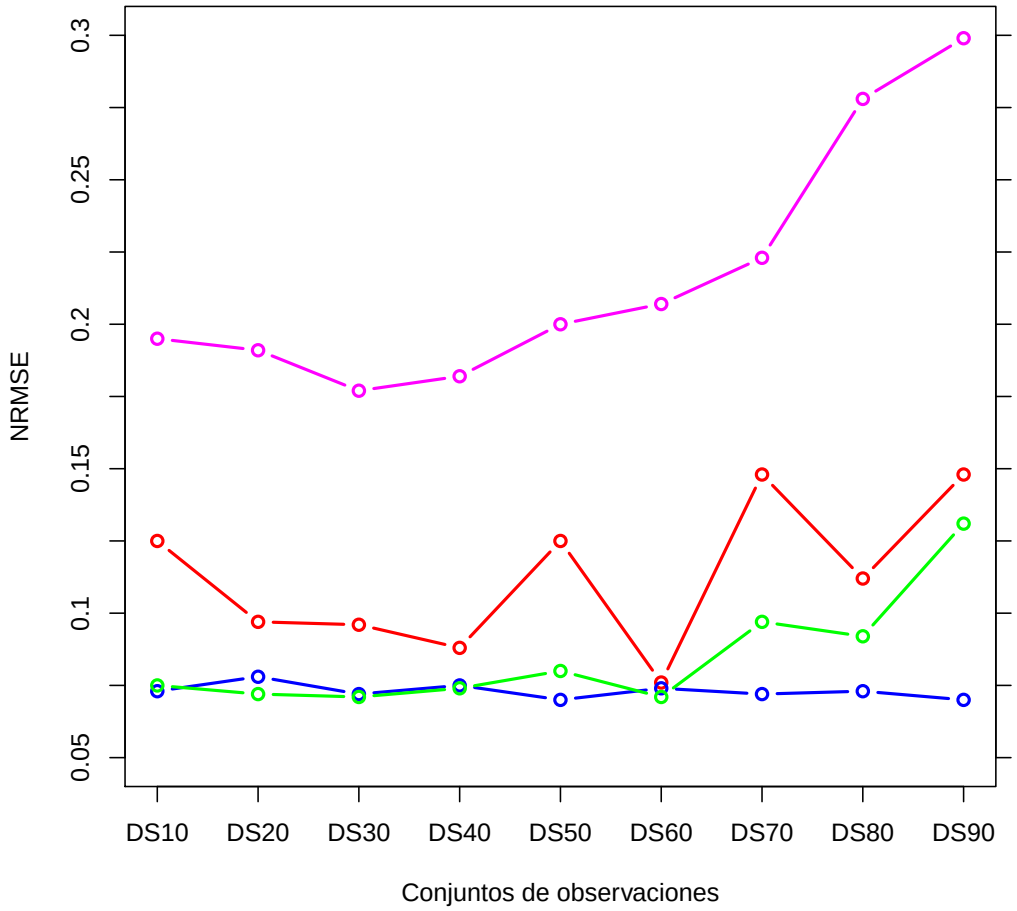


Figura 5.5: NRMSE de los métodos de solución al inferir la concentración de cobre al variar el tamaño del conjunto de observaciones. En rojo se muestran los resultados de ANN, en magenta los correspondientes a GP, en azul aquellos de IGMM y en verde los pertenecientes a BTSR.

Métodos de	Conjuntos de observaciones								
solución	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
ANN	0.125	0.097	0.096	0.088	0.125	0.076	0.148	0.112	0.148
GP	0.195	0.191	0.177	0.182	0.200	0.207	0.223	0.278	0.299
IGMM	0.073	0.078	0.072	0.075	0.070	0.074	0.072	0.073	0.070
BTSR	0.075	0.072	0.071	0.074	0.080	0.071	0.097	0.092	0.131

Tabla 5.5: NRMSE de los métodos de solución al inferir la concentración de cobre al variar el tamaño del conjunto de observaciones.

absoluto (MAE) de IGMM en comparación con los métodos estudiados en [2] se presenta en las Figuras 5.6 y 5.7.

Como puede observarse, IGMM presenta un error medio absoluto considerablemente menor que el resto de los métodos estudiados en [2] para ambas tareas, por encima de los procesos Gaussianos completos, lo cual puede considerarse como evidencia de la capacidad de IGMM para problemas complejos de regresión tanto para problemas de tarea única como para problemas multitarea, en este caso múltiples salidas correlacionadas de forma no-trivial.

En la siguiente sección se presentan las conclusiones generales que se derivan de este estudio.

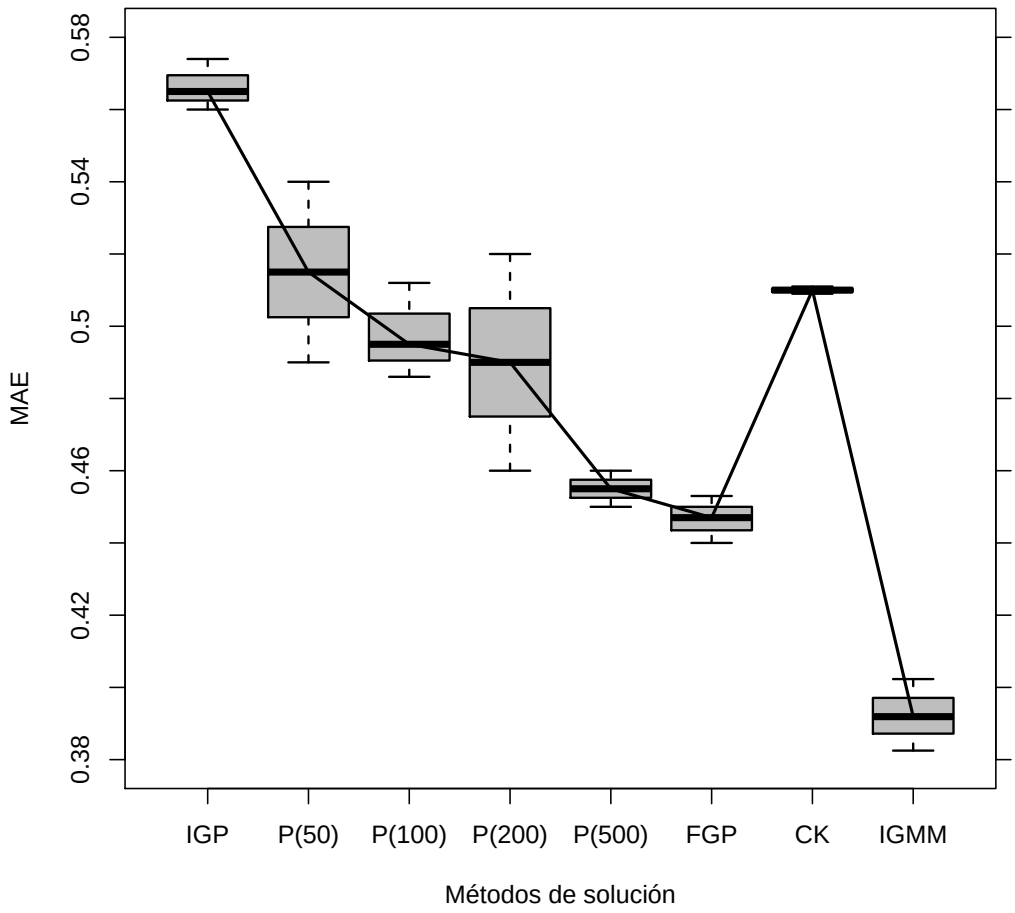


Figura 5.6: Error medio absoluto y desviación estándar del pronóstico de cadmio para 10 repeticiones de IGMM, además de procesos Gaussianos independientes (IGP), aproximaciones condicionales parcialmente independientes con M valores inductores ($P(M)$), procesos Gaussinos completos (FGP) y *cokriging* ordinario (CK).

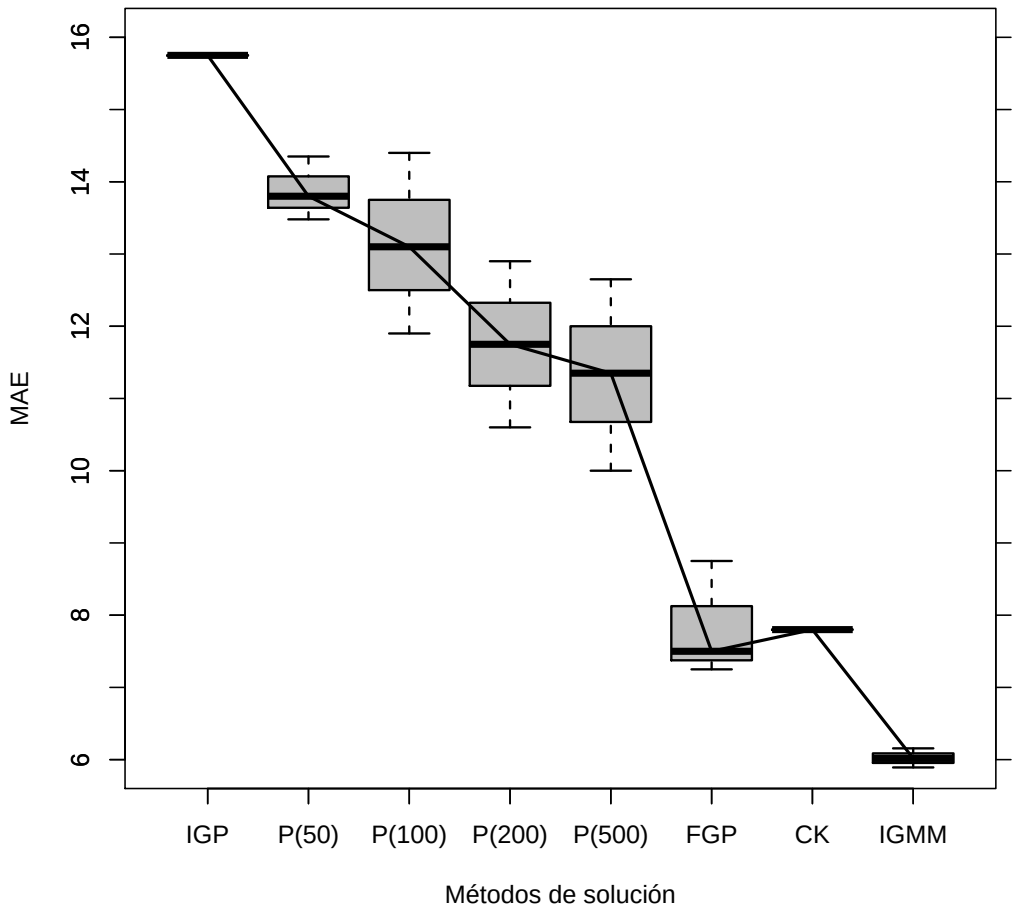


Figura 5.7: Error medio absoluto y desviación estándar del pronóstico de cobre para 10 repeticiones de IGMM, además de procesos Gaussianos independientes (IGP), aproximaciones condicionales parcialmente independientes con M valores inductores ($P(M)$), procesos Gaussinos completos (FGP) y *cokriging* ordinario (CK).

Capítulo 6

Conclusiones

Finalmente, en este capítulo se discuten y analizan las conclusiones derivadas de los resultados de este estudio. En la Sección 6.1 se presentan las conclusiones generales de esta tesis, mientras que en la Sección 6.2 se resumen las contribuciones que se aportan por medio de ésta. Por último, la Sección 6.3 describe el trabajo a futuro de este estudio.

6.1 Conclusiones

En esta tesis se estudió el desempeño en la capacidad de pronóstico para dos métodos Bayesianos reconocidos, como lo son las redes neuronales y los procesos Gaussianos, y de un método que representa algunas ventajas por encima de éstos pero que no ha sido aplicado de manera continua en la literatura: el modelo de mezclas infinitas de Gaussianas. Adicionalmente, el desempeño de los tres métodos Bayesianos es comparado con el método *bootstrap*, un método que ha ganado popularidad en los últimos años al ser capaz de resolver problemas en donde otros métodos han fallado. El impacto que tiene el tamaño del conjunto de entrenamiento en el desempeño de estos métodos es también estudiado, especialmente la variación en la capacidad de pronóstico al disminuir la cantidad de observaciones para el entrenamiento del modelo.

Se mostró que el modelo de mezcla infinita de Gaussianas es una técnica general robusta en cuanto al tamaño del conjunto de entrenamiento, destacándose de entre

los demás métodos de solución por obtener las mejores capacidades de pronóstico en todos los problemas de estudio y con todos los conjuntos de entrenamiento. La diferencia promedio en la capacidad de pronóstico es considerable en el problema del XOR continuo, mientras que esta diferencia promedio con respecto al resto de los métodos parece no ser considerable en la inferencia de la afinidad de los acoplamientos enzimáticos, aunque para ciertos tamaños muestrales sí representa una marcada ventaja, como es el caso de DS7. Una cuestión similar sucede para la inferencia de concentración de metales pesados, donde además IGMM muestra una capacidad de pronóstico considerablemente mejor en comparación con algoritmos del estado del arte presentados en [2]. Los métodos Bayesianos contienen de manera implícita un mecanismo que previene el sobreajuste, y esto ha quedado evidenciado dado que los tres métodos Bayesianos presentan NRMSEs relativamente pequeños, sin embargo, el *bootstrap* ha demostrado ser una herramienta adecuada para evitar el sobreajuste, al arrojar resultados comparables a aquellos del modelo de mezcla infinita de Gaussianas, e incluso capacidades de pronóstico que superan las de un método clásico como lo es la red neuronal. Comparando las redes neuronales con el modelo de mezcla infinita de Gaussianas, se encuentra que existe una importante diferencia entre la capacidad de pronóstico de ambos métodos cuando el tamaño del conjunto de entrenamiento es pequeño, aminorándose esta diferencia conforme se aumenta el tamaño muestral. No obstante, se muestra que los cuatro métodos de estudio son capaces de modelar adecuadamente el ruido y la no-linealidad de los conjuntos de observaciones multidimensionales, previniendo el sobreajuste. Además de la robustez empírica que presenta, entre las ventajas de utilizar el modelo de mezcla infinita de Gaussianas se encuentran: (i) el modelo es capaz de realizar clasificación y regresión sin modificar su estructura, (ii) no tiene parámetros que necesiten de un ajuste y (iii) no necesita información externa a priori de sus parámetros para realizar inferencia Bayesiana. Nuestro interés en este método consiste en que es un algoritmo completamente automático que aprende eficazmente la cantidad de componentes y las Gaussianas que modelan sus observaciones, además de asignar un grado de *responsabilidad* que tiene cada Gaussiana para una observación. Esta es una propiedad sumamente interesante

para el aprendizaje en línea.

Para llevar a cabo estos objetivos se implementaron computacionalmente los tres métodos Bayesianos en el lenguaje *R*, y se seleccionaron problemas de estudio que permitieran un análisis interpretativo de la variación en la capacidad de pronóstico de los métodos. Los tres problemas presentados en este estudio son problemas no-lineales retadores provenientes de diferentes disciplinas, teniendo características diferentes. El problema XOR continuo tiene como característica el hecho de que sus observaciones carecen de ruido, mientras que en el problema de la afinidad de acoplamientos enzimáticos existen altos niveles de ruido y no-linealidad. El problema de concentración de metales pesados contiene, además de ruido y no-linealidad, correlación en sus atributos de salida, lo que representa una disminución en la capacidad de pronóstico cuando las tareas se hacen individualmente, en comparación con un enfoque multitarea.

6.2 Contribuciones

Las contribuciones que se derivan de este estudio se enlistan a continuación:

- Se comprobó la efectividad de los métodos Bayesianos bajo estudio. Se comprobó además un método no-Bayesiano que, de acuerdo a su popularidad y su capacidad para resolver problemas en donde otros métodos fallan, logró un desempeño comparable al modelo de mezcla infinita de Gaussianas y en ocasiones superior a las redes neuronales.
- Se demostró el potencial que tienen los métodos Bayesianos, especialmente la mezcla infinita de Gaussianas, para evitar el sobreajuste conforme el tamaño muestral decrece.
- Se aplicó el modelo infinito de mezcla de Gaussianas a problemas retadores de interés actual, con lo que se enriquece el bajo número de aplicaciones de este método que han sido estudiadas.

- Se implementaron las redes neuronales y los procesos Gaussianos con aprendizaje Bayesiano en R, al estar sólo disponibles en otros lenguajes.

- Se implementó el modelo de mezcla infinita de Gaussianas, cuya versión multivariada no se encuentra disponible en ningún lenguaje, hasta donde nosotros sabemos. Esta implementación es de importancia para estudios que se derivan de éste y de otros que son ajenos, por ejemplo, en materia de aprendizaje en línea.

- Se escribió un apéndice que detalla la implementación computacional de los métodos Bayesianos, una aportación importante para comprender la forma de operar del modelo de mezcla infinita de Gaussianas y para esquivar las complicaciones que implica su implementación. Esta contribución es importante, al no haber un documento que detalle tal implementación.

6.3 Trabajo futuro

Este estudio presenta una gran área de oportunidades en que la investigación puede extenderse. Algunos estudios que puede derivarse de éste son:

- Incorporar al estudio otros métodos no-lineales cuya mejor característica sea prevenir el sobreajuste.

- Realzar estudios acerca de los estados metaestables de las cadenas Markov en los problemas de estudio, que disminuyen la capacidad de pronóstico.

- Extender las comparativas con modelos Bayesianos lineales eficientes encontrados en literatura, para determinar los límites de predicción lineales.

- Abarcar una mayor cantidad de problemas de estudio, buscando en éstos diferentes características que permitan proporcionar resultados intuitivos y de gran calidad.

- Mejorar la capacidad de pronóstico de los métodos para problemas en que se tiene una correlación en los atributos de salida, incorporando a los modelos un intercambio de información que proporcione mejores resultados que el resolver los problemas secuencialmente.

Apéndice A

Implementación Computacional

El objetivo principal de esta tesis es aplicar métodos Bayesianos estadísticos y de aprendizaje automático en problemas retadores de estudio como lo son el XOR continuo, la afinidad en acoplamientos enzimáticos y la concentración de metales pesados en la capa superficial del suelo, conforme se disminuye el tamaño del conjunto de entrenamiento, de modo que sea posible analizar la pérdida de precisión que exhiben los métodos como efecto que tiene el tamaño muestral, así como el ruido y la no-linealidad presentes en las observaciones. De igual forma, se desea analizar el desempeño de las técnicas Bayesianas en comparación con una técnica no-Bayesiana que ha ganado popularidad en los últimos años, el *bootstrap*, una herramienta estadística que produce conjuntos de observaciones adicionales a partir de la muestra original con el fin de estimar la distribución muestral de un estadístico. En este apéndice se presenta una descripción de la implementación computacional de los métodos de solución propuestos. Esta descripción consiste en las ecuaciones necesarias para implementar cada método, así como una explicación de los pasos a seguir. Se presentan también a lo largo del apéndice algunas observaciones y sugerencias para evitar problemas en la implementación (especialmente por precisión numérica). En la Sección A.1 se describe la implementación de las redes neuronales, mientras que en la Sección A.2 se detallan los procesos Gaussianos. Finalmente, en la Sección A.3 se describe el modelo de mezclas infinitas de Gaussianas.

A.1 Redes Neuronales Artificiales

De manera inicial se genera un conjunto aleatorio de pesos para la red neuronal, muestreando de una distribución uniforme en el intervalo $[-1, 1]$:

$$p(\boldsymbol{w}) \sim \mathcal{U}[-1, 1]. \tag{A.1}$$

Posteriormente se generan muestras para los pesos que provengan de su distribución posterior dadas las observaciones, utilizando para esto el muestreo de Gibbs con paso Metrópolis descrito en la Sección 2.2.2. El parámetro de escala se ajusta de tal forma que se acepten una cantidad entre el 30 y el 70 por ciento de los candidatos. La distribución posterior de interés es proporcional a la verosimilitud de los pesos dadas las observaciones (i.e., la probabilidad de las observaciones dados los pesos). Por cuestiones numéricas, el muestreo se hace sobre la función logarítmica de la distribución posterior:

$$\ln p(\boldsymbol{w}|\boldsymbol{X}_N, \boldsymbol{T}_N) \propto -\frac{1}{2\sigma_v^2} \sum_{n=1}^{N} \sum_{k=1}^{u} \left[t_n^k - y_k^n(\boldsymbol{x}_n; \boldsymbol{w}_k)\right]^2 - \frac{N}{2} \ln(2\pi\sigma_v^2). \tag{A.2}$$

Una vez que se han tomado M_B muestras de los pesos se procede a descartar las muestras pertenecientes a la etapa inicial transitoria de la cadena de Markov generada, como se detalla en 2.2.1. Al final de este proceso, se tiene un conjunto conformado por M muestras. Ahora bien, dada una observación $\boldsymbol{x}_f$ para la cual se desean inferir los objetivos en $\boldsymbol{t}_f$, se computan M salidas de la red neuronal utilizando cada una de las M muestras de los pesos. La k-ésima salida de la m-ésima red neuronal dada la observación n se evalúa mediante:

$$y_k^n(\boldsymbol{x}_n; \boldsymbol{w}_k^m) = \sum_{j=1}^{q} w_{hy}^{jk} \cdot \tanh\left[\sum_{i=1}^{p} w_{eh}^{ij} \cdot x_n^i + w_h^j\right] + w_y^k \qquad \forall k, m, n. \tag{A.3}$$

Finalmente, el valor esperado a la salida de la red neuronal para una observación $\boldsymbol{x}_f$ consiste en promediar sobre las M salidas:

$$\langle \boldsymbol{t}_f | \boldsymbol{x}_f \rangle = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{y}^f(\boldsymbol{x}_f; \boldsymbol{w}^m). \tag{A.4}$$

A.2 Procesos Gaussianos

El método comienza con la selección de valores iniciales para los parámetros $\boldsymbol{\theta} = (\sigma_f^2, \sigma_v^2, l_1, \ldots, l_p)^T$. A falta de información preliminar, estos valores iniciales se muestrean uniformemente del intervalo $[0.1, 10]$:

$$p(\boldsymbol{\theta}) \sim \mathcal{U}[0.1, 10]. \tag{A.5}$$

De acuerdo al Teorema de Bayes (Teorema 2.1.1), el conjunto más probable de los parámetros, $\boldsymbol{\theta}_{MP}$, corresponde con aquel que maximiza la función logarítmica de la verosimilitud marginal de los parámetros. De esta forma, se aplica el método del recocido simulado, presentado en la Sección 2.3, para resolver el siguiente problema de optimización:

$$\boldsymbol{\theta}_{MP} = \text{máx} \left[-\frac{u}{2} \ln |\boldsymbol{C}_N| - -\frac{1}{2} \sum_{k=1}^{u} (\boldsymbol{t}^k)^T \boldsymbol{C}_N^{-1} \boldsymbol{t}^k - \frac{Nu}{2} \ln(2\pi) \right], \tag{A.6}$$

en donde la matriz de covarianza $\boldsymbol{C}_N$, con dimensión $(N \times N)$, se obtiene aplicando la función exponencial cuadrada para cada una de las N observaciones. El elemento nn' de esta matriz se desarrolla mediante:

$$C_N(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) = \sigma_f^2 \exp \left[-\frac{1}{2} \sum_{i=1}^{p} \frac{(x_n^i - x_{n'}^i)^2}{l_i^2} \right] + \delta \sigma_v^2 \qquad \forall n, n', \tag{A.7}$$

donde δ representa una delta de Dirac y u la dimensionalidad de los atributos de salida.

Posteriormente, dada una observación $\boldsymbol{x}_f$ para la cual se desean inferir los objetivos en $\boldsymbol{t}_f$, se computa el vector de covarianza $\boldsymbol{c_{XF}}$, con dimensión $(N \times 1)$, aplicando la misma función para cada una de las N observaciones y la observación $\boldsymbol{x}_f$. El n-ésimo elemento de este vector se desarrolla mediante:

$$c_{XF}(\boldsymbol{x}_n, \boldsymbol{x}_f) = \sigma_f^2 \exp\left[-\frac{1}{2}\sum_{i=1}^{p}\frac{(x_n^i - x_f^i)^2}{l_i^2}\right] + \delta\sigma_v^2 \qquad \forall n. \tag{A.8}$$

Finalmente, el valor esperado del objetivo t_f^k dada la observación $\boldsymbol{x}_f$ está dado por:

$$\widehat{t}_f^{\,k} = \boldsymbol{c}_{XF}^T \boldsymbol{C}_N^{-1} \boldsymbol{t}^k \qquad \forall k. \tag{A.9}$$

Para poder definir las barras de error de esta predicción se necesita evaluar la función exponencial cuadrada en la observación $\boldsymbol{x}_f$, lo que proporciona un escalar:

$$c_f(\boldsymbol{x}_f, \boldsymbol{x}_f) = \sigma_f^2 \exp\left[-\frac{1}{2}\sum_{i=1}^{p}\frac{(x_f^i - x_f^i)^2}{l_i^2}\right] + \delta\sigma_v^2. \tag{A.10}$$

Así, las barras de error pueden evaluarse mediante:

$$\boldsymbol{\Sigma}_{\widehat{t}_f^{\,k}} = c_f - \boldsymbol{c}_{XF}^T \boldsymbol{C}_N^{-1} \boldsymbol{c}_{XF}^T \qquad \forall k. \tag{A.11}$$

A.3 Mezcla Infinita de Gaussianas

El algoritmo se inicializa con un componente. En esta sección se asume que las observaciones son multivariadas, con dimensión d. Dado que no se hace una distinción entre atributos de entrada y salida, el conjunto de observaciones está conformado por $\mathcal{D} = \{\boldsymbol{y}_n\}_{n=1}^N$, donde a su vez $\boldsymbol{y}_n = (y_n^1, \ldots, y_n^d)^T$. La media y la precisión inicial de este único componente corresponden con la media y precisión de las observaciones, respectivamente:

$$\boldsymbol{\mu} = \boldsymbol{\mu}_{\mathcal{D}}, \tag{A.12}$$

$$\boldsymbol{S} = \boldsymbol{\Sigma}_{\mathcal{D}}^{-1}. \tag{A.13}$$

El resto de los parámetros e hiperparámetros del modelo toman valores iniciales de acuerdo a un muestreo de sus distribuciones posteriores:

$$\boldsymbol{\lambda} = \mathcal{N}(\boldsymbol{\mu}_{\mathcal{D}}, \boldsymbol{\Sigma}_{\mathcal{D}}), \tag{A.14}$$

$$\boldsymbol{R} = \mathcal{W}(d, \boldsymbol{\Sigma}_{\mathcal{D}}^{-1}), \tag{A.15}$$

$$\boldsymbol{W} = \mathcal{W}(d, \boldsymbol{\Sigma}_{\mathcal{D}}), \tag{A.16}$$

$$\beta = 1/\mathcal{G}(1, 1/d) + d - 1, \tag{A.17}$$

$$\alpha = 1/\mathcal{G}(1, 1), \tag{A.18}$$

$$I_c = 1, \tag{A.19}$$

en donde I_c es un indicador de la última observación asignada a una clase no representada. Ahora bien, el muestreo de las distribuciones posteriores puede ser en cualquier orden, pero el hecho de incorporar y eliminar componentes hace del orden de muestreo una difícil decisión. Para hacer que el muestreo funcione de manera veloz y que exista una convergencia a la distribución posterior deseada se deberían muestrear todas las variables en $\boldsymbol{c}$ de manera simultánea, pero esto implica la adición y remoción de clases, lo que a su vez afecta el resto de los parámetros en $\boldsymbol{c}$. Sin embargo, cuando no hay componentes que se adicionan o remueven, el resto de los parámetros no se ven afectados. Una posible opción es muestrear un parámetro en $\boldsymbol{c}$ por ciclo de Gibbs, pero esto ocasionaría muestras altamente correlacionadas en el tiempo, lo que sería ineficiente. El lado opuesto sería formar un ciclo para muestrear los parámetros $\boldsymbol{c}$ en que se adicionen o remuevan tantas clases como sea necesario, hasta terminar de muestrear todos los parámetros efectivamente. Esto no es computacionalmente adecuado, ya que tal ciclo sería demasiado lento. En esta tesis se sigue

la sugerencia de Mandel [38] para balancear la eficiencia del código y la velocidad de convergencia. Esta sugerencia consiste en muestrear tantos elementos de $\boldsymbol{c}$ como sea posible hasta la primera adición de una clase. Una vez que se adiciona una clase se muestrean el resto de los parámetros para luego continuar con el elemento $c_n = I_c$.

Muestreo de $\boldsymbol{\mu}_j$. A partir de este punto se comienza con el muestreo de Gibbs como se describió en la Sección 2.2.1, muestreando una variable por vez mediante las distribuciones posteriores correspondientes. El muestreo de Gibbs inicia extrayendo una muestra para la media del componente j mediante:

$$p(\boldsymbol{\mu}_j|\boldsymbol{c}, \mathcal{D}, \boldsymbol{S}_j, \boldsymbol{\lambda}, \boldsymbol{R}) \sim \mathcal{N}\left((h_j \bar{\boldsymbol{y}}_j \boldsymbol{S}_j + \boldsymbol{\lambda}\boldsymbol{R})\boldsymbol{\Sigma}_j^{\mu}, \boldsymbol{\Sigma}_j^{\mu}\right) \qquad \forall j, \tag{A.20}$$

en donde la matriz de covarianza es:

$$\boldsymbol{\Sigma}_j^{\mu} = (h_j \boldsymbol{S}_j + \boldsymbol{R})^{-1} \qquad \forall j. \tag{A.21}$$

La variable h_j representa aquí la cantidad de observaciones que pertenecen a la clase j. Por su parte, $\bar{\boldsymbol{y}}_j$ representa la media de tales observaciones:

$$\bar{\boldsymbol{y}}_j = \frac{1}{h_j} \sum_{i:c_n=j} \boldsymbol{y}_i. \tag{A.22}$$

Muestreo de $\boldsymbol{\lambda}$ y $\boldsymbol{R}$. Posteriormente se muestrean los hiperparámetros de las medias de los componentes a través de sus distribuciones posteriores:

$$p(\boldsymbol{\lambda}|\boldsymbol{\mu}, \boldsymbol{R}) \sim \mathcal{N}\left(\left(\boldsymbol{\mu}_{\mathcal{D}}\boldsymbol{\Sigma}_{\mathcal{D}}^{-1} + \sum_{j=1}^{k} \boldsymbol{\mu}_j \boldsymbol{R}\right)\boldsymbol{\Sigma}_j^{\lambda}, \boldsymbol{\Sigma}_j^{\lambda}\right), \tag{A.23}$$

$$p(\boldsymbol{R}|\boldsymbol{\mu}, \boldsymbol{\lambda}) \sim \mathcal{W}\left(k+1, \left[\frac{1}{k+1}\left(\boldsymbol{\Sigma}_{\mathcal{D}} + \sum_{j=1}^{k}(\boldsymbol{\mu}_j - \boldsymbol{\lambda})^T(\boldsymbol{\mu}_j - \boldsymbol{\lambda})\right)\right]^{-1}\right). \tag{A.24}$$

donde la matriz de covarianza $\boldsymbol{\Sigma}_j^{\lambda}$ es:

$$\boldsymbol{\Sigma}_j^{\lambda} = (\boldsymbol{\Sigma}_{\mathcal{D}}^{-1} + k\boldsymbol{R})^{-1}. \tag{A.25}$$

Muestreo de $\boldsymbol{S}_j$. A continuación se muestrean las precisiones de los componentes mediante:

$$p(\boldsymbol{S}_j|\boldsymbol{c}, \mathcal{D}, \boldsymbol{\mu}_j, \beta, \boldsymbol{W}) \sim \mathcal{W}(\beta + h_j, \boldsymbol{V}) \quad \forall j, \tag{A.26}$$

donde la matriz de escala de la distribución Wishart es:

$$\boldsymbol{V} = \left[\frac{1}{\beta + h_j}\left(\beta\ \boldsymbol{W} + \sum_{i:c_i=j}(\boldsymbol{y}_i - \boldsymbol{\mu}_j)^T(\boldsymbol{y}_i - \boldsymbol{\mu}_j)\right)\right]^{-1} \tag{A.27}$$

Se recomienda utilizar la pseudo-inversa de Moore-Penrose [48] en vez de la inversión ordinaria para $\boldsymbol{V}$, para evitar que el código regrese un error cuando ésta tienda a ser computacionalmente no-singular debido a cuestiones de precisión numérica. La pseudo-inversa de Moore-Penrose es una generalización de la inversa ordinaria para matrices no-singulares. Este fenómeno aparece con mayor o menor frecuencia dependiendo del problema que se desea resolver.

Muestreo de $\boldsymbol{W}$. Como siguiente paso, se muestrea la matriz $\boldsymbol{W}$ de las precisiones de los componentes mediante:

$$p(\boldsymbol{W}|\boldsymbol{S}_1, \ldots, \boldsymbol{S}_k, \beta) \sim \mathcal{W}\left(k\beta + 1, \left[\frac{1}{k\beta + 1}\left(\boldsymbol{\Sigma}_{\mathcal{D}}^{-1} + \beta\sum_{j=1}^{k}\boldsymbol{S}_j\right)\right]\right). \tag{A.28}$$

Muestreo de β. Posteriormente, para muestrear β se aplica el muestreo de Gibbs con paso Metrópolis, descrito en la Sección 2.2.2, para generar muestras para g mediante la distribución posterior:

$$\begin{aligned}\ln p(g|\boldsymbol{S}_1,\dots\boldsymbol{S}_k,\boldsymbol{W}) \;\propto\; & -\frac{3}{2}\ln\left(g+d-1\right)\,\frac{(g+d-1)k}{2}\ln|\boldsymbol{W}| \\ & +\left(\frac{d}{2(g+d-1)}\right)\left(\frac{g+d-1}{2}\right)^{(g+d-1)kd/2} \\ & +\sum_{j=1}^{k}\frac{|\boldsymbol{S}_j|^{\frac{g}{2}-1}\exp\left(-\dfrac{(g+d-1)\;\mathrm{tr}(\boldsymbol{W}\boldsymbol{S}_j)}{2}\right)}{\displaystyle\prod_{i=0}^{d-1}\Gamma\left(\frac{g+i}{2}\right)}. \qquad \text{(A.29)}\end{aligned}$$

Las muestras para β se recuperan recordando que $\beta = g + d - 1$. El parámetro de escala se ajusta de tal forma que se acepten una cantidad entre el 30 y el 70 por ciento de los candidatos. El proceso para generar muestras para β presenta una de las dificultades encontradas al implementar el modelo. Dado que β se utiliza el parámetro de los grados de libertad de la distribución Wishart, debe cumplir con una restricción de esta distribución:

$$\beta \geq d. \qquad \text{(A.30)}$$

Debido a esto, se decide generar una cantidad suficiente de muestras para g, eliminar aquellas en que $g < 1$ y seleccionar aleatoriamente una de tales muestras. Luego, la muestra de β es $\beta = g + d - 1$.

Muestreo de α. De manera semejante, se aplica el muestreo de Gibbs con paso Metrópolis para generar muestras de α mediante:

$$\ln p(\alpha|k,N) = (k-3/2)\ln\alpha - \frac{1}{2\alpha}\ln\Gamma(\alpha) - \ln\left(\Gamma(N+\alpha)\right). \qquad \text{(A.31)}$$

No existe ninguna restricción para el valor de α, por lo que basta con tomar como muestra el primer candidato que sea aceptado (después de ajustar el parámetro de escala).

Distribución posterior de las clases no-representadas. A continuación, se evalúa la probabilidad posterior para las clases mediante el producto de la verosimilitud de las clases dadas las observaciones y la distribución a priori de los indicadores $\boldsymbol{c}$. La distribución posterior de las clases depende del número de observaciones asociadas a la clase. Si $I_c \leq N$ entonces existen clases no-representadas, por lo que es necesario proponer $(N - I_c)$ clases, una por cada observación no asociada a un componente. Las medias y precisiones se muestrean mediante sus distribuciones a priori:

$$\boldsymbol{\mu}_q = \mathcal{N}(\boldsymbol{\lambda}, \boldsymbol{R}^{-1}), \qquad \forall q \in \{1, \ldots, N - I_c\} \tag{A.32}$$

$$\boldsymbol{S}_q = \mathcal{W}(\beta, \boldsymbol{W}^{-1}) \qquad \forall q, \tag{A.33}$$

mientras que la verosimilitud sigue una distribución Gaussiana:

$$p(\boldsymbol{y}_n|\boldsymbol{\mu}_q, \boldsymbol{S}_q) = \mathcal{N}(\boldsymbol{y}_n|\boldsymbol{\mu}_q, \boldsymbol{S}_q^{-1}) \qquad \forall n, q. \tag{A.34}$$

Sea $\boldsymbol{p}^{\text{n-r}}$ el vector de las N probabilidades posteriores de las clases no-representadas. Cada elemento de este vector corresponde con:

$$p_n^{\text{n-r}} = \begin{cases} \left[\dfrac{\alpha}{N - 1 + \alpha} \times \mathcal{N}(\boldsymbol{y}_n|\boldsymbol{\mu}_n, \boldsymbol{S}_n^{-1})\right], & \text{si } I_c \leq n \leq N \text{ y } 0 < h_{n',j}, \\ 0, & \text{en cualquier otro caso.} \end{cases}$$

Conforme el muestreo de Gibbs avanza aparecen algunos problemas numéricos en el muestreo del hiperparámetro $\boldsymbol{R}$, cuya matriz tiende a ser no-singular y no-simétrica debido a cuestiones de precisión computacional, lo que ocasiona que el código devuelva un error. Para las cuestiones de no-singularidad de la matriz se recomienda utilizar la pseudo-inversa de Moore-Penrose en vez de la inversión ordinaria para $\boldsymbol{R}$, mientras que para las cuestiones de no-simetría se recomienda verificar y corregir en este punto la falta de simetría numérica de la matriz $\boldsymbol{R}$ muestreada.

Distribución posterior de las clases representadas. De manera similar, sea $\boldsymbol{L}$ la matriz de verosimilitud para las clases representadas, con dimensión $(k \times N)$. El elemento L_{jn} de esta matriz corresponde con la verosimilitud del componente j dada la n-ésima observación:

$$L_{jn} = \mathcal{N}(\boldsymbol{y}_n | \boldsymbol{\mu}_j, \boldsymbol{S}_j^{-1}) \qquad \forall j, n. \tag{A.35}$$

Si $\boldsymbol{A}$ es la matriz de probabilidad a priori para las clases representadas, con dimensión $(k \times N)$, entonces el elemento A_{jn} de esta matriz está en función de la cantidad de observaciones diferentes a $\boldsymbol{y}_n$ que la clase j tiene asociada:

$$A_{jn} = \begin{cases} \dfrac{h_{n',j}}{(N-1+\alpha)}, & \text{si } h_{n',j} > 0, \\ \dfrac{1}{(N-1+\alpha)}, & \text{si } h_{n',j} = 0 \text{ y } c_n = j. \end{cases}$$

Así, la matriz de probabilidad posterior de las clases establecidas es el producto algebraico de la matriz de verosimilitud y la matriz de probabilidad a priori:

$$\boldsymbol{P}^{\mathrm{r}} = \boldsymbol{A} \cdot \boldsymbol{L}, \tag{A.36}$$

en donde $(\cdot)$ indica una multiplicación de elementos. Mediante la distribución multinomial se muestrea una clase para cada observación:

$$\boldsymbol{c}_n = \mathcal{M}\left(\begin{bmatrix} \boldsymbol{p}^{\text{n-r}} \\ \boldsymbol{P}^{\mathrm{r}} \end{bmatrix}\right) = \mathcal{M}\left(\begin{bmatrix} p_1^{\text{n-r}} & \cdots & p_N^{\text{n-r}} \\ P_{jn}^{\mathrm{r}} & \cdots & P_{jN}^{\mathrm{r}} \\ \vdots & \ddots & \vdots \\ P_{kn}^{\mathrm{r}} & \cdots & P_{kN}^{\mathrm{r}} \end{bmatrix}\right) \qquad \forall j. \tag{A.37}$$

Finalmente, se asignan las observaciones a las nuevas clases, desde $n = I_c$ hasta la primera adición de una nueva clase. De modo que los elementos en $\boldsymbol{c}$ desde $n = I_c$ hasta encontrar una clase que tenga una observación cuyo componente corresponda

a una clase no representada se reemplazan por los elementos correspondientes en $\boldsymbol{c}_n$. Las clases ya establecidas que se encuentran en $\boldsymbol{c}$ se mantienen, mientras que aquellas que no tienen observaciones asociadas se eliminan. Las clases propuestas que fueron muestreadas se incorporan al modelo. Cuando $I_c > N$, entonces se reinicia mediante $I_c = 1$. Este procedimiento representa un ciclo de Gibbs.

Bibliografía

[1] Albert, M., «Bayesian Rationality and Decision Making: A Critical Review», *Analyse & Kritik*, **2003**(25), págs. 101–117, 2003.

[2] Álvarez, M. A. y N. D. Lawrence, «Computationally Efficient Convolved Multiple Output Gaussian Processes», *J. Mach. Learn. Res.*, págs. 1459–1500, julio 2011, URL `http://dl.acm.org/citation.cfm?id=2021026.2021048`.

[3] Atteia, O., J. P. Dubois y R. Webster, «Geostatistical analysis of soil contamination in the Swiss Jura.», *Environmental Pollution*, **86**(3), págs. 315–327, 1994, URL `http://www.ncbi.nlm.nih.gov/pubmed/15091623`.

[4] Attias, H., «A Variational Bayesian Framework for Graphical Models», en *In Advances in Neural Information Processing Systems 12*, MIT Press, págs. 209–215, 2000.

[5] Barber, D., *Bayesian Reasoning and Machine Learning*, Cambridge University Press, febrero 2011, URL `http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0521518148`.

[6] Barnett, S., *Matrix Methods for Engineers and Scientists*, MacGraw-Hill, 1979.

[7] Bartlett, M. S., *An introduction to stochastic processes, with special reference to methods and applications*, tercera edición, Cambridge University Press, Cambridge ; New York :, 1978.

[8] BERTSEKAS, D. P. y D. P. BERTSEKAS, *Nonlinear Programming*, segunda edición, Athena Scientific, septiembre 1999, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/1886529000.

[9] BISHOP, C. M., *Pattern recognition and machine learning*, primera edición, Springer, octubre 2006.

[10] BOX, G. E. P. y G. JENKINS, *Time Series Analysis, Forecasting and Control*, Holden-Day, Incorporated, 1990.

[11] CANTY, A. y B. D. RIPLEY, *boot: Bootstrap R (S-Plus) Functions*, r package version 1.3-4, 2012.

[12] CAWLEY, G. C. y N. L. C. TALBOT, «Preventing Over-Fitting during Model Selection via Bayesian Regularisation of the Hyper-Parameters», *J. Mach. Learn. Res.*, **8**, págs. 841–861, May 2007, URL http://dl.acm.org/citation.cfm?id=1248659.1248690.

[13] CHOPIN, N., «Jim Albert: Bayesian computation with R», *Statistics and Computing*, **19**(1), págs. 111–112, marzo 2009, URL http://dx.doi.org/10.1007/s11222-008-9069-8.

[14] COX, D. R., D. COMMENGES, A. C. DAVISON, P. J. SOLOMON y S. R. WILSON, «The Oxford Dictionary of Statistical Terms», en Y. Dodge (editor), *The Oxford Dictionary of Statistical Terms*, Oxford University Press, 2003.

[15] DAVISON, A. C. y D. V. HINKLEY, *Bootstrap Methods and Their Applications*, Cambridge University Press, Cambridge, iSBN 0-521-57391-2, 1997, URL http://statwww.epfl.ch/davison/BMA/.

[16] EFRON, B. y R. J. TIBSHIRANI, *An Introduction to the Bootstrap*, Chapman & Hall, New York, 1993.

[17] ELISSEEFF, A. y M. PONTIL, «Leave-one-out Error and Stability of Learning Algorithms with Applications», en J. Suykens, , G. Horvath, S. Basu, C. Mic-

chelli y J. Vandewalle (editores), *Learning Theory and Practice*, NATO ASI Series, IOS Press, Amsterdam; Washington, DC, 2002.

[18] FOX, J., «Bootstrapping Regression Models Appendix to An R and S-PLUS Companion to Applied Regression», , 2002.

[19] FREEDMAN, D., R. PISANI y R. PURVES, *Statistics, 4th Edition*, cuarta edición, W. W. Norton, marzo 2007, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0393929728.

[20] GEMAN, S., E. BIENENSTOCK y R. DOURSAT, «Neural networks and the bias/variance dilemma», *Neural Comput.*, **4**, págs. 1–58, January 1992, URL http://dl.acm.org/citation.cfm?id=148061.148062.

[21] GILKS, W. R., S. RICHARDSON y D. SPIEGELHALTER, *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics (Chapman & Hall/CRC Interdisciplinary Statistics)*, primera edición, Chapman & Hall/CRC, diciembre 1995, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0412055511.

[22] GOOVAERTS, P., *Geostatistics for natural resources evaluation*, Oxford University Press, USA, 1997.

[23] HÄRDLE, W. y L. SIMAR, *Applied Multivariate Statistical Analysis*, Springer, septiembre 2003, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/3540030794.

[24] HASTIE, T., R. TIBSHIRANI y J. H. FRIEDMAN, *The Elements of Statistical Learning*, corrected edición, Springer, julio 2003, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0387952845.

[25] HAYKIN, S., *Neural Networks: A Comprehensive Foundation*, Macmillan, New York, 1994.

[26] HECHT-NIELSEN, R., «Theory of the backpropagation neural network», en *Neural Networks, 1989. IJCNN., International Joint Conference on*, págs. 593–605, junio 1989, URL http://dx.doi.org/10.1109/IJCNN.1989.118638.

[27] HUBER, P., *Robust Statistics*, Wiley, New York, 1974.

[28] JEFFERYS, W. H. y J. O. BERGER, «Ockham's razor and Bayesian analysis», *American Scientist*, **80**(January-February), págs. 64–72, 1992.

[29] JOAQUIN QUI, N.-C. y C. E. RASMUSSEN, «A Unifying View of Sparse Approximate Gaussian Process Regression», *J. Mach. Learn. Res.*, **6**, págs. 1939–1959, 2005, URL http://portal.acm.org/citation.cfm?id=1194909.

[30] KHALIL, H. K., *Nonlinear Systems (3rd Edition)*, tercera edición, Prentice Hall, diciembre 2001, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0130673897.

[31] KIRKPATRICK, S., C. D. GELATT y M. P. VECCHI, «Optimization by Simulated Annealing», *Science, Number 4598, 13 May 1983*, **220, 4598**, págs. 671–680, 1983, URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.18.4175.

[32] KOHAVI, R., «A study of cross-validation and bootstrap for accuracy estimation and model selection», en *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, págs. 1137–1143, 1995, URL http://dl.acm.org/citation.cfm?id=1643031.1643047.

[33] KONONENKO, I., M. ROBNIK-SIKONJA, M. ROBNIK y U. POMPE, «ReliefF for estimation and discretization of attributes in classification, regression, and ILP problems», , 1996.

[34] LIU, L., D. M. HAWKINS, S. GHOSH y S. S. YOUNG, «Robust singular value decomposition analysis of microarray data», *Proc Natl Acad Sci U S*

A, **100**(23), págs. 13 167–13 172, 2003, URL http://www.pnas.org/content/100/23/13167.abstract.

[35] LOHR, S. L., *Sampling: Design and Analysis*, primera edición, Duxbury Press, diciembre 1999, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0534353614.

[36] MACCHIARULO, A., I. NOBELI y J. M. THORNTON, «Ligand selectivity and competition between enzymes in silico.», *Nat Biotechnol*, **22**(8), págs. 1039–1045, agosto 2004, URL http://dx.doi.org/10.1038/nbt999.

[37] MACKAY, D. J. C., *Information Theory, Inference and Learning Algorithms*, primera edición, Cambridge University Press, junio 2003.

[38] MANDEL, M., «Implementing the Infinite GMM.», *Final project in Prof. Tony Jebara's machine Learning course, Columbia University*, 2005.

[39] MCLACHLAN, G. y D. PEEL, *Finite Mixture Models*, primera edición, Wiley Series in Probability and Statistics, Wiley-Interscience, octubre 2000.

[40] MELKUMYAN, A. y F. RAMOS, «Multi-Kernel Gaussian Processes.», en *IJCAI'11*, págs. 1408–1413, 2011.

[41] MONTGOMERY, D., E. PECK y G. VINING, *Introduction to linear regression analysis*, tercera edición, Wiley series in probability and statistics, Wiley, New York, NY [u.a.], 2001, URL http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+321916239&sourceid=fbw_bibsonomy.

[42] NEAL, R. M., *Bayesian Learning for Neural Networks (Lecture Notes in Statistics)*, primera edición, Springer, agosto 1996, URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0387947248.

[43] NEAL, R. M., «Markov Chain Sampling Methods for Dirichlet Process Mixture Models», *Journal of Computational and Graphical Statistics*, **9**(2), págs. 249–265, 2000, URL http://www-clmc.usc.edu/~{}cs599_ct/neal-TR1998.pdf.

[44] NELSON, D. L. y M. M. COX, *Lehninger Principles of Biochemistry, Fourth Edition*, fourth edition edición, 2004.

[45] NGUYEN, V.-A., Z. KOUKOLÍKOVÁ-NICOLA, F. BAGNOLI y P. LIÓ, «Bayesian Inference on Hidden Knowledge in High-Throughput Molecular Biology Data», en *Proceedings of the 10th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence*, PRICAI '08, Springer-Verlag, Berlin, Heidelberg, págs. 829–838, 2008, URL http://dx.doi.org/10.1007/978-3-540-89197-0_77.

[46] NGUYEN, V.-A., Z. KOUKOLÃKOVÃ¡-NICOLA, F. BAGNOLI y P. LIO, «Noise and non-linearities in high-throughput data», *Journal of Statistical Mechanics: Theory and Experiment*, **2009**(01), pág. P01 014, 2009, URL http://stacks.iop.org/1742-5468/2009/i=01/a=P01014.

[47] PARSONS, S., «Introduction to Machine Learning by Ethem Alpaydin, MIT Press, 0-262-01211-1, 400 pp.,», *Knowl. Eng. Rev.*, **20**, págs. 432–433, December 2005, URL http://dl.acm.org/citation.cfm?id=1132846.1132849.

[48] PETKOVIĆ, M. D. y P. S. STANIMIROVIĆ, «Iterative method for computing the Moore-Penrose inverse based on Penrose equations», *J. Comput. Appl. Math.*, **235**, págs. 1604–1613, January 2011, URL http://dx.doi.org/10.1016/j.cam.2010.08.042.

[49] PLAGNOL, V. y S. TAVARE, *Approximate Bayesian Computation and MCMC*, Springer Verlag, págs. 99–113, 2003.

[50] R DEVELOPMENT CORE TEAM, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0, 2011, URL http://www.R-project.org/.

[51] RABUNAL, J. y J. DORADO, *Artificial neural networks in real-life applications*, Idea Group Publishing, USA, 2006.

[52] RASMUSSEN, C. E., «The Infinite Gaussian Mixture Model», en *In Advances in Neural Information Processing Systems 12*, tomo 12, págs. 554–560, 2000, URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.9111.

[53] RASMUSSEN, C. E. y C. WILLIAMS, *Gaussian Processes for Machine Learning*, MIT Press, 2006, URL http://www.gaussianprocess.org/gpml/.

[54] RIPLEY, B. D., *Pattern Recognition and Neural Networks*, Cambridge University Press, enero 1996.

[55] SEIFFERT, U., «Multiple Layer Perceptron Training Using Genetic Algorithms», , 2001.

[56] SHAWE-TAYLOR, J., R. HOLLOWAY, P. L. BARTLETT, R. C. WILLIAMSON y M. ANTHONY, «Structural Risk Minimization over Data-Dependent Hierarchies», , 1996.

[57] SHIN, H. y S. CHO, «Pattern Selection for Support Vector Classifiers», en *Proceedings of the Third International Conference on Intelligent Data Engineering and Automated Learning*, IDEAL '02, Springer-Verlag, London, UK, UK, págs. 469–474, 2002, URL http://dl.acm.org/citation.cfm?id=646288.686626.

[58] SHIN, H. y S. CHO, «Fast pattern selection for support vector classifiers», en *Proceedings of the 7th Pacific-Asia conference on Advances in knowledge discovery and data mining*, PAKDD'03, Springer-Verlag, Berlin, Heidelberg, págs. 376–387, 2003, URL http://dl.acm.org/citation.cfm?id=1760894.1760944.

[59] TAKAKE Y., O. Y. y S. T., «Approximations of nonlinear functions by feedforward neural networks.», *Proceedings of the Japan Classification Society Meeting*, págs. 26–33, 1994.

[60] TIERNEY, L. y J. B. KADANE, «Accurate Approximations for Posterior Moments and Marginal Densities», *Journal of the American Statistical Associa-*

tion, **81**(393), págs. 82–86, marzo 1986, URL http://dx.doi.org/10.2307/2287970.

[61] TIPPING, M. E. y C. M. BISHOP, «Probabilistic Principal Component Analysis», *Journal of the Royal Statistical Society, Series B*, **61**, págs. 611–622, 1999, URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.33.4726.

[62] TROYANSKAYA, O., M. CANTOR, G. SHERLOCK, P. BROWN, T. HASTIE, R. TIBSHIRANI, D. BOTSTEIN y R. B. ALTMAN, «Missing value estimation methods for DNA microarrays.», *Bioinformatics (Oxford, England)*, **17**(6), págs. 520–525, junio 2001, URL http://dx.doi.org/10.1093/bioinformatics/17.6.520.

[63] WALPOLE, R. E., R. H. MYERS, S. L. MYERS y K. YE, *Probability & statistics for engineers and scientists*, 8ª edición, Pearson Education, Upper Saddle River, 2007.

[64] WALSH, B., «Markov Chain Monte Carlo and Gibbs Sampling», , 2004, URL http://web.mit.edu/wingated/www/introductions/mcmc-gibbs-intro.pdf.

[65] WEHRENS, R., H. PUTTER y L. M. BUYDENS, «The bootstrap: a tutorial», *Chemometrics and Intelligent Laboratory Systems*, **54**(1), págs. 35–52, diciembre 2000, URL http://dx.doi.org/10.1016/S0169-7439(00)00102-7.

[66] WILSON, A. G., D. A. KNOWLES y Z. GHAHRAMANI, «Gaussian Process Regression Networks», , octubre 2011, 1110.4411, URL http://arxiv.org/abs/1110.4411.

[67] WINKLER, R. L., *An introduction to Bayesian inference and decision / Robert L. Winkler*, Holt, Rinehart and Winston, New York :, 1972.

[68] WITTEN, I. H. y E. FRANK, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series*

in Data Management Systems), primera edición, Morgan Kaufmann, octubre 1999.

[69] XIANGYANG LIU, H. G., «Hyperbolic tangent function based two layers structure neural network.», *Electronics and Optoelectronics (ICEOE), 2011 International Conference on*, **4**, págs. V4–376–V4–379, 2011.

[70] YAO, T., «Nonparametric cross-covariance modeling as exemplified by soil heavy metal concentrations from the Swiss Jura», *Geoderma*, **88**, págs. 13–38, 1998.

[71] ZHOU, X., X. WANG y E. R. DOUGHERTY, «Missing-value estimation using linear and non-linear regression with Bayesian gene selection.», *Bioinformatics*, **19**(17), págs. 2302–2307, noviembre 2003, URL `http://view.ncbi.nlm.nih.gov/pubmed/14630659`.

Ficha autobiográfica

Mario Alberto Saucedo Espinosa

Candidato para el grado de Maestro en Ciencias
con especialidad en Ingeniería de Sistemas

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

Métodos Bayesianos estadísticos y de aprendizaje automático para estimación en sistemas complejos

Nací el 02 de noviembre de 1983, en la ciudad de Monterrey, Nuevo León, México, siendo el primer hijo del Ing. Mario A. Saucedo de los Santos y la Sra. Adriana M. Espinosa de Saucedo, y hermano mayor de José Guillermo y María Catalina. Mis estudios primarios y secundarios transcurrieron sin nada especial que valga la pena recordar en estas líneas. Fue hasta el año 2002 cuando tuve mi primer contacto con la Universidad Autónoma de Nuevo León (UANL), lo que a la postre se convertiría en un romance que no culmina con la redacción de este proyecto. Después de un par de años de indiferencia académica me decidí a estudiar Ingeniería Química como un reto más en mi vida. Fue así que entré a la Facultad de Ciencias Químicas en el año 2003, lo que a la larga recordaría como una de las mejores etapas

de mi vida, siendo reconocido como Premio al Mérito Académico y titulándome como Ingeniero Químico en el año 2008. Después de un par de años laborando en proyectos de investigación y como docente, me decidí a continuar mis estudios profesionales a nivel posgrado, ingresando al Posgrado en Ingeniería de Sistemas de la UANL a inicios del año 2010, donde nuevamente mi vida daría un giro muy agradable. Fue ahí que tuve la oportunidad de dar pláticas en congresos nacionales e internacionales, de publicar mis investigaciones, de conocer Europa a través de una estancia académica de investigación, y sobre todo, de conocer a mi novia, Brenda Ayala. Este manuscrito representa el fin de esa otrora nueva meta, fungiendo como opción para conseguir el título de Maestro en Ciencias de la Ingeniería de Sistemas.