ESCUELA DE INGENIERÍA DE FUENLABRADA

GRADO EN INGENIERÍA BIOMÉDICA

TRABAJO FIN DE GRADO

APRENDIZAJE AUTOMÁTICO PARA LA DETECCIÓN DE RESISTENCIAS ANTIMICROBIANAS A PARTIR DE DATOS DE ESPECTROMETRÍA DE MASAS

Autor: Leticia Medina García

Tutor: Cristina Soguero Ruíz

Co-tutor: Jesús Jiménez Ibáñez

Curso académico 2022/2023

El fracaso es simplemente la oportunidad de empezar de nuevo, esta vez de forma más inteligente.

– Henry Ford

Agradecimientos

En primer lugar, me gustaría agradecer a Luis Mancera y Cristina Soguero la oportunidad de conocer Clover desde dentro, aumentando mis conocimientos durante los meses de prácticas externas y el transcurso de este trabajo. En este periodo, me he sentido acogida y apoyada en todo momento, además de tener la suerte de aprender con Jesús, Gema y Manuel, quienes no han dudado en ayudarme siempre que lo he necesitado.

A aquellos profesores que han despertado mi interés en la programación y en especial a Cristina, por iniciarme en el estudio de la Inteligencia Artificial y sus múltiples funcionalidades. También por no dudar en tutorizar este trabajo dándome la oportunidad de llevarlo a cabo con su ayuda, y demostrarme que el esfuerzo es imprescindible para conseguir el éxito.

Esta etapa llega a su fin y, a pesar de los duros momentos, me quedo con sentimientos positivos acerca de estos 4 años. Gracias a mis compañeras, que puedo llamar amigas, porque han conseguido facilitar y motivar los momentos más difíciles. Y por supuesto, la gran suerte de haber podido realizar mi Erasmus en Finlandia. Fueron meses de incertidumbre y continuo aprendizaje que me permitieron vivir experiencias increíbles y conocer a personas maravillosas.

Por último a mi familia, mi gran apoyo. Gracias mamá y papá por no poner límites a nuestros sueños y no dudar de nuestras habilidades, gracias por facilitarnos las herramientas para que consigamos el futuro que deseamos tener. Gracias Carlos por acompañarme en este camino y entenderme. Gracias Felipe por no dudar nunca de mis capacidades y apoyarme en todo momento, haciéndolo todo más fácil y estando siempre. Gracias, en general, a mi familia y amigos por formar parte de mi y ser un respaldo en todo momento, permitiéndome desconectar siempre que lo necesitaba y dándome el espacio necesario cuando no.

Resumen

Las resistencias antimicrobianas (RAM) representan una de las mayores amenazas a la salud pública mundial en la actualidad. Su impacto es tan significativo que se estima que, en el año 2050, podrían causar alrededor de 10 millones de muertes en todo el mundo si no se invierte la tendencia. Por ello, es imprescindible abordar este problema con métodos innovadores que permitan un diagnóstico rápido y preciso de las infecciones, reduciendo así la mortalidad asociada a ellas y minimizando el surgimiento de nuevas resistencias.

Una técnica rápida, precisa y asequible para la identificación de microorganismos es la espectrometría de masas de desorción/ionización láser asistida por matriz acoplada a un detector de tiempo de vuelo (MALDI-TOF-MS), que al combinarlo con técnicas de aprendizaje automático, aumenta su potencial. El uso de estos modelos permite aprovechar los espectros generados por la espectrometría de masas, lo que posibilita discriminar los microorganismos resistentes a los antimicrobianos de los sensibles, y la identificación de biomarcadores asociados a la RAM.

Con el propósito de lograr esta meta, se ha definido un modelo de clasificación denominado Red Neuronal Bayesiana (BNN) en el que se han optimizado hiperparámetros como la cantidad de neuronas ocultas o la función de activación para conseguir un algoritmo con buena generalización. Esta técnica se ha implementado con dos conjuntos de datos diferentes resultantes tras la técnica MALDI-TOF-MS. En primer lugar, se toman las muestras de *Aspergillus fumigatus* y se discriminan la especie *Sensu Stricto* (s.s.) de las especies crípticas y las muestras de *Aspergillus fumigatus s.s.* resistentes de las sensibles a los azoles. A continuación, se toman las muestras de bacterias *E. Coli* con el objetivo de discernir los organismos con respuesta resistente y sensible a los antibióticos Ciprofloxacin, Ceftriaxone y Cefepime.

Los resultados adquiridos con la BNN se comparan con los de los modelos Random Forest y LightGBM, obteniéndose en el caso del conjunto de *Aspergillus fumigatus* mejores prestaciones de la BNN. Sin embargo, en el conjunto de *E. Coli*, aunque semejantes, la BNN obtiene unos valores inferiores a LightGBM en las prestaciones *balanced accuracy* y AUC, y superiores en el caso de *precision* y *specificity* para la clasificación de las muestras resistentes y sensibles a los antibióticos Ciprofloxacin y Ceftriaxone. En la discriminación en función de la respuesta al antibiótico Cefepime, tanto la BNN como LightGBM obtienen las mismas prestaciones, superando las de Random Forest.

Por lo tanto, la BNN diseñada demuestra ser capaz de distinguir correctamente las categorías definidas del conjunto *Aspergillus fumigatus*, facilitando a los profesionales clínicos la toma de decisiones en este caso. No obstante, en las muestras de *E. Coli* las categorías obtenidas están distanciadas de la realidad, no siendo la BNN adecuada para este propósito.

Índice general

Índice de figuras

Índice de tablas

Lista de acrónimos y abreviaturas

AA Aprendizaje Automático

ANN *Artificial Neural Network*

AUC *Area Under the Curve*

BNN *Bayesian Neural Network*

CAF Cuestionario de Actividades Funcionales

CDC Centros para el Control y la Prevención de Enfermedades

CV *Cross-Validation*

ECDC Centro Europeo para la Prevención y el Control de las Enfermedades

EMA Agencia Europea del Medicamento

FN Falso Negativo

FP Falso Positivo

FTIR Infrarrojos por Transformada de Fourier

GPU *Graphics Processing Unit*

IA Inteligencia Artificial

IR *Infrared Spectrometry*

LightGBM *Light Gradient Boosting Machine*

MALDI-TOF MS *Matrix-Assisted Laser Desorption/Ionization Time of Flight Mass Spectrometry*

ML *Machine Learning*

MS *Mass Spectrometry*

OMS Organización Mundial de la Salud

PCA *Principal Component Analysis*

RAM Resistencia a los Antimicrobianos

RF *Random Forest*

ROC *Receiver Operating Characteristic*

SARM *Staphylococcus Aureus* resistente a la Meticilina

SNR Relación Señal a Ruido

TFG Trabajo Fin de Grado

TFP Tasa de Falsos Positivos

TIC Corriente Iónica Total

TOF *Time of Flight*

TVP Tasa de Verdaderos Positivos

VIH Virus de Inmunodeficiencia Humana

VN Verdadero Negativo

VP Verdadero Positivo

Capítulo 1

Introducción y objetivos

Este capítulo constituye una presentación del problema de la resistencia antimicrobiana, incluyendo sus consecuencias sanitarias y económicas, y la importancia de combatirlo. A continuación, se definen los objetivos del presente Trabajo de Fin de Grado (TFG), la correspondiente metodología seguida y una breve estructura de la memoria.

1.1. Contexto y motivación

El descubrimiento de la Penicilina en 1928 por Alexander Fleming, supuso un cambio revolucionario en la medicina moderna [8]. En la década de 1940, los antibióticos comenzaron a utilizarse para el tratamiento de infecciones graves. La Penicilina, sobretodo, tuvo un gran éxito en la Segunda Guerra Mundial para asistir a los soldados. No obstante, en apenas 10 años, las bacterias comenzaron a ser resistentes a este antibiótico, convirtiendo esta oposición en un grave problema clínico, puesto que el fármaco suponía la base de muchos tratamientos. Con el fin de paliar el problema, se desarrollaron nuevos antibióticos, como Meticilina o Vancomicina, que combatieran las infecciones, pero con el trascurso de los años fueron creando nuevas resistencias de las bacterias [8]. En consecuencia, en las dos últimas décadas se han desarrollado muy pocos antimicrobianos, lo que está limitando su campo de acción y demandando en la actualidad nuevas clases de antibióticos que combatan las infecciones [9].

La resistencia a los antibióticos en particular, y a los antimicrobianos en general, definida como la capacidad del organismo para sobrevivir en concentraciones de antimicrobiano que inhiben a otros de la misma especie [10], supone un problema mundial en la pérdida de eficacia de los fármacos. Este hecho se debe a varias causas, destacando el uso excesivo e inadecuado

del tratamiento, incluyéndose en este punto las prescripciones del medicamento erróneas que suponen entre el 30 y el 50 % de los casos [8]. Además, se estima que el 50 % de las prescripciones médicas de antibióticos en hospitales se llevan a cabo sin conocer adecuadamente si se trata de una infección y de su tipo [11], o que entre el 30 y el 60 % de los antibióticos prescritos en las unidades de cuidados intensivos en atención hospitalaria son innecesarios, inadecuados o subóptimos [12]. Las mutaciones y el intercambio horizontal de genes también se incluye como causa de la resistencia a antibióticos [10].

Las infecciones potencialmente mortales causadas por *Enterobacteriaceae* suelen ser resistentes a los antibióticos Carbapenem, por lo que la Colistina es el tratamiento de último recurso para estas infecciones [8]. Sin embargo, en los últimos años se ha detectado resistencia a este tratamiento. Esta familia de bacterias comprende a organismos como *Klebsiella Pneumoniae* y *Escherichia Coli*, que son resistentes a los antibióticos Carbapenem [9] y Fluoroquinolonas [13] respectivamente en al menos el 50 % de los pacientes. Además, se calcula que los pacientes infectados por SARM (*Staphylococcus Aureus* resistente a la Meticilina) tienen un 64 % más de posibilidades de morir que los pacientes infectados por la bacteria no resistente [8].

El Centro Europeo para la Prevención y el Control de las Enfermedades (ECDC) declaró en 2007 a la resistencia a los antimicrobianos como una de las amenazas de mayor peso para las enfermedades infecciosas [14]. El ECDC junto con la Agencia Europea del Medicamento (EMA) estimaron que en Europa, al menos 25 mil pacientes mueren cada año debido a este suceso [15].

En el año 2015, la Organización Mundial de la Salud (OMS) publicó un informe en el que se identifican afecciones médicas comunes (la tuberculosis, la malaria, las enfermedades de transmisión sexual, las infecciones del tracto urinario, la neumonía, las infecciones del torrente sanguíneo y las intoxicaciones alimentarias) como resistentes a un gran espectro de medicamentos antimicrobianos [9]. Asimismo, en 2016, se publicó el informe O' Neill [16], encargado por el Gobierno británico, en el que se predijo que en 2050 la resistencia a los antimicrobianos (RAM) podría convertirse en la principal causa de muerte mundial con un 10 millones de fallecimientos al año, superando a los 8,2 millones de personas que mueren al año por cáncer [16]. Además, la mejora del acceso a los medicamentos y la ausencia de políticas antibióticas han contribuido en el uso indebido de antibióticos por parte de países como Brasil, Rusia, India, China y Sudáfrica [16].

El gran coste económico debido a la RAM es también importante de mencionar. Estas infecciones provocan enfermedades graves que pueden derivar en ingresos hospitalarios prolongados y en el uso de diversos tratamientos para encontrar el más adecuado. En Europa se estima un

gasto de más de nueve mil millones de euros al año, y según los Centros para el Control y la Prevención de Enfermedades (CDC), en Estados Unidos supone 20.000 millones de dólares en costes sanitarios directos y 35.000 millones anuales debidos a la pérdida de productividad [17].

La RAM requieree un riesgo en la atención y sistemas sanitarios modernos ya que estos fármacos son la base de una gran variedad de tratamientos como el cáncer, la cirugía y el trasplante de órganos, además de implicar un alto coste económico. Ante este evidente problema, encontrar una solución se ha convertido en una prioridad para los gobiernos y organismos mundiales [9]. La OMS propuso el ”Plan de acción mundial sobre la resistencia a los antimicrobianos” en 2015 con el objetivo principal de mantener una capacidad constante de tratamiento y prevención de enfermedades infecciosas con medicamentos eficaces y seguros, accesible para todos. Además, mostró la importancia de la vigilancia y la investigación en el tratamiento de las infecciones para aumentar el conocimiento sobre ellas [18].

En el contexto del conocimiento sobre epidemiología y prevalencia, el uso de los modelos de Aprendizaje Automático (AA) o *Machine Learning* (ML) en inglés, cada vez es más usual para predecir la resistencia a distintos antibióticos en patógenos basándose en el contenido génico y la composición del genoma. Además, basando estas herramientas en espectrometría de masas (MS) de ionización láser asistida por matriz acoplada a un detector de tiempo de vuelo (MALDI-TOF), se puede aplicar en la optimización del tratamiento y la administración de antibióticos [19]. La función principal de la empresa CLOVER Bioanalytical Software S.L.U. está relacionada con esta tarea. Clover se encarga del tratamiento de datos para la identificación microbiana y la química bioanalítica, ocupándose también de aplicaciones de diagnóstico por diferentes tipos de espectrometría de masas [20]. Su plataforma online, Clover MS Data Analysis Software [2], permite la clasificación de microorganismos resistentes.

En el presente TFG se colabora con Clover, utilizando los datos de microorganismos proporcionados por ellos en la definición de un modelo de ML, capaz de determinar si ciertos microbios son resistentes a partir de sus masas espectrales tomadas con la técnica MALDI-TOF MS. Se manejan conjuntos de muestras del hongo *Aspergillus fumigatus* y de la bacteria *E. Coli*, de las que se conoce los espectros de masas obtenidos en el procedimiento MALDI-TOF MS, componiendo el vector de características de cada muestra. Los resultados de este algoritmo se comparan con los modelos de la plataforma online de Clover para valorar si el nuevo modelo puede implementarse en ella.

1.2. Objetivos y metodología

El objetivo principal de este TFG consiste en implementar, optimizar, entrenar y validar un modelo de Red Neuronal Bayesiana, BNN (del inglés *Bayesian Neural Network*) que será comparado con otros algoritmos de ML que incorpora Clover MS Data Analysis Software, con la finalidad de ser utilizado en la plataforma. El modelo de ML creado se realiza para la detección de resistencias antimicrobianas en datos clínicos provenientes de la técnica MALDI-TOF MS. Para lograr este objetivo, se llevan a cabo las siguientes etapas:

- Conocer el problema de los microorganismos patógenos capaces de resistir a los antimicrobianos que pretenden combatirlos.
- Analizar de forma exploratoria los datos proporcionados por la empresa Clover Biosoft en su plataforma para conocer en detalle sus características. Éstos son espectros de masas que han sido preprocesados en la plataforma previamente.
- Tratar los conjuntos de datos aportados para poder diseñar el modelo de aprendizaje automático BNN, que, principalmente, permite clasificar las muestras en resistentes o sensibles a unos determinados antifúngicos y antibióticos según el tipo de microorganismo al que pertenezcan las muestras.
- Reducir la dimensionalidad del vector de características seleccionando generando nuevas variables representativas mediante el Análisis de Componentes Principales, PCA (del inglés *Principal Component Analysis*).
- Investigar acerca de los modelos de BNN y desarrollar un algoritmo de este tipo capaz de predecir la situación de resistencia de las muestras a los antimicrobianos correspondientes. En esta etapa se toman el 70 % de los ejemplos totales para entrenar la red y encontrar los mejores hiperparámetros del modelo.
- Validar la BNN con el 30 % de las observaciones en cada conjunto de datos.
- Diseñar los algoritmos *Random Forest* (RF) y LightGBM en la plataforma de Clover, incluyendo la misma división de las muestras en *train* y *test* que se utiliza en BNN, incluyendo la búsqueda de los hiperparámetros más adecuados en cada modelo.
- Comparar los resultados obtenidos en la plataforma con los adquiridos tras la predicción con la BNN, evaluando si el nuevo modelo es apto para incluirse en la plataforma de Clover.

En la Figura 1.1 se presenta un diagrama de Gantt que desglosa, de manera orientativa, las tareas realizadas durante el desarrollo del presente TFG, tratando de aproximar las fechas en las que se realizaron las tareas lo máximo posible a la realidad.

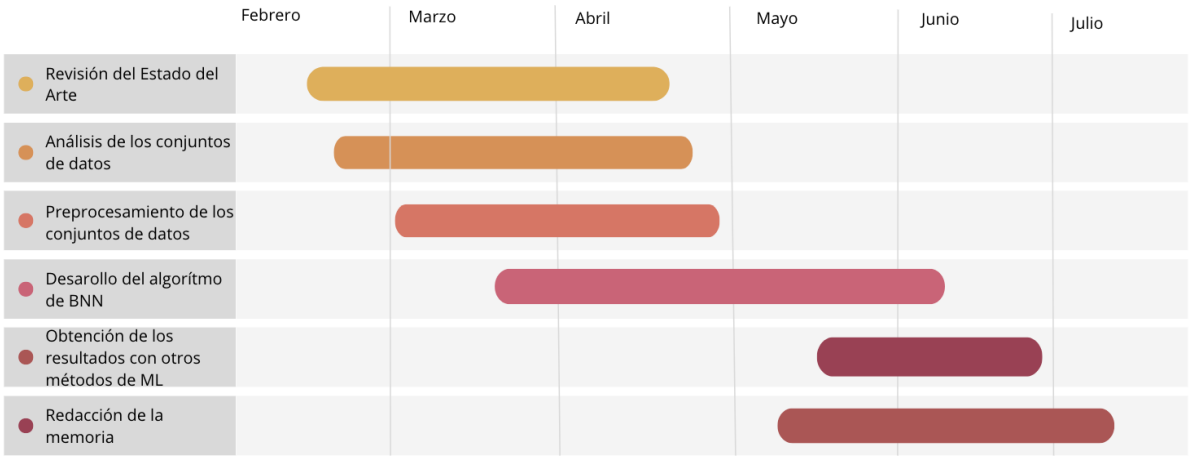


Figura 1.1: Diagrama de Gantt.

1.3. Estructura de la memoria

A continuación se describe el contenido incluido en cada capítulo de la memoria:

- **Capítulo 1: Introducción y objetivos**. Capítulo inicial en el que se tratan tres secciones distintas. En la primera se contextualiza el tema tratado en este trabajo justificando su elección. La segunda trata los objetivos y la metodología usada para lograrlos. Y finalmente, en la tercera se presenta la estructura empleada en la memoria de este trabajo.
- **Capítulo 2: Conceptos previos**. Este capítulo está dividido en cinco secciones en las que se presentan y describen los conceptos necesarios para la comprensión de este TFG. Concretamente se tratan la resistencias a los antimicrobianos, la técnica MALDI-TOF MS, microorganismos como *Aspergillus fumigatus* y *E. Coli* y la plataforma Clover MS Data Analysis Software.
- **Capítulo 3: Métodos**. Trata de describir los modelos de ML empleados, como BNN, RF y LightGBM, además de presentar algunos conceptos clave para el desarrollo de éstos.
- **Capítulo 4: Base de datos y análisis descriptivo**. Este capítulo pretende describir los datos proporcionados por la empresa Clover Biosoft y el procesamiento de éstos.

- **Capítulo 5: Experimentos y resultados**. Se define el modelo BNN creado y se exponen los resultados tras el proceso de clasificación las muestras con los tres modelos de ML requeridos en este trabajo.
- **Capítulo 6: Conclusiones y líneas futuras**. Se resumen los resultados obtenidos durante la etapa de experimentación, se comparan y se exponen las conclusiones derivadas de ellos. Finalmente, se presentan posibles líneas futuras que podrían desarrollarse a partir de este estudio.

Capítulo 2

Conceptos previos

En este capítulo se introducen una serie de conceptos fundamentales que ayudan a entender el marco clínico y biológico sobre el que se desarrolla este TFG.

2.1. Resistencias antimicrobianas

Un antimicrobiano es definido como una sustancia que elimina, inhibe o detiene el crecimiento de bacterias, hongos y parásitos [21]. Según el microorganismo al que ataca, éste puede ser antibiótico, antifúngico, antivírico o antiprotozoario. El antimicrobiano se utiliza para la prevención y el tratamiento de infecciones debido a estos microbios en seres humanos, animales y plantas [21].

La resistencia de las bacterias a los antibióticos supone una gran amenaza para la atención sanitaria, incluyendo el alto riesgo en el tratamiento de enfermedades infecciosas comunes, procedimientos médicos y cirugía mayor o el aumento del coste de la atención médica entre otros. En consecuencia, estos patógenos generan un incremento significativo en la morbimortalidad de los pacientes hospitalizados, afectando especialmente a los pacientes más vulnerables situados en las unidades de cuidados intensivos, neonatología u oncología [17,22]. Según un estudio publicado en la revista médica *The Lancet*, esta resistencia a los antibióticos, y antimicrobianos en general, supone una de las principales causas de muerte a nivel mundial, superando a la infección por el virus de la inmunodeficiencia humana y provocando más de 1.2 millones de muertes en 2019 mundialmente [23]. Incluso se estima que las infecciones por resistencia a los antimicrobianos causan más de 700 000 muertes al año y provocarán al menos 10 millones de muertes en el año 2050 si no se invierte esta tendencia [24].

La RAM es el proceso por el cual bacterias, virus, hongos y parásitos cambian a lo largo del tiempo y dejan de responder a los medicamentos antimicrobianos que pretenden combatirlos [25, 26]. Esto dificulta el tratamiento de las infecciones y aumenta el riesgo de propagación de enfermedades leves, graves y la muerte. En consecuencia, los medicamentos antimicrobianos pierden su eficacia y las infecciones persisten en el cuerpo incrementando el riesgo de propagación [26]. La causa principal de esta resistencia es el cambio genético de los organismos de forma natural, aunque el uso excesivo y erróneo de los antimicrobianos ayudan a acelerar el proceso [23].

Es por ello por lo que es esencial tratar la infección de forma temprana con un antimicrobiano efectivo y además, prevenir el desarrollo de esta resistencia. Sin embargo, las técnicas actuales basados en cultivos pueden llevar hasta 72 horas para obtener resultados [24], lo que supone una demora significativa para tomar decisiones clínicas y para proporcionar antimicrobianos efectivos. Durante este intervalo de tiempo, el paciente puede recibir antimicrobianos de amplio espectro, favoreciendo la aparición de resistencias, o antimicrobianos de espectro estrecho, no siendo suficientemente eficaces para combatir al microorganismo [27].

En este contexto, un método rápido preciso y asequible para identificar microorganismos es la espectrometría de masas MALDI-TOF [28], que se explicará en la siguiente Sección con mayor detalle. Así, la aplicación del ML a los datos de espectros de masas tomados con esta técnica puede representar una nueva y significativa herramienta que mejora la optimización del tratamiento y la administración de antimicrobianos [19].

2.2. MALDI-TOF MS

La espectrometría de masas de tiempo de vuelo con ionización por desorción láser asistida por matriz (MALDI-TOF MS, del inglés *Matrix-Assisted Laser Desorption/Ionization - Time of Flight*) es un método preciso, rápido y rentable para la caracterización e identificación microbiana en diversas áreas incluyendo el diagnóstico médico, la biodefensa y el control de calidad en alimentos [1].

Un procedimiento sencillo con esta técnica consiste en tomar la muestra del microorganismo y colocarla en la placa objetivo de MALDI-TOF MS. A la placa se agrega una matriz cocristalizante compuesta por un ácido orgánico, como el ácido ferúlico, el ácido sináptico o el ácido α-ciano-4-hidroxicinámico entre otras [1, 29], con el objetivo de mejorar la calidad del espectro de masas que se generará. A continuación, se introduce la placa objetivo en el interior del espectrómetro de masas, donde la muestra recibe unos pulsos cortos de láser en los puntos

que se van a analizar, ionizando las moléculas microbianas y acelerándose en el vacío mediante un campo eléctrico [1]. Las partículas más ligeras viajan más rápido que las más pesadas, generándose un espectro de masas que refleja la cantidad de iones que llegan al detector a lo largo del tiempo. Por lo tanto, el tiempo que tarda en llegar cada partícula desde su origen hasta el detector se conoce como tiempo de vuelo (TOF) y depende de la masa m y la carga z de la partícula. El espectro de masas resultante, representa la proporción entre masa y carga m/z en el eje de abscisas y la intensidad, que depende de la cantidad de partículas que alcanzan el detector por cada valor de m/z, en el eje de ordenadas (véase en la Figura 2.1). Este espectro resultante es una huella del microbiano que se compara con las bases de datos existentes, que se van actualizando y contiene especies comunes y raras con importancia clínica [30], para identificar al organismo en una especie [1].

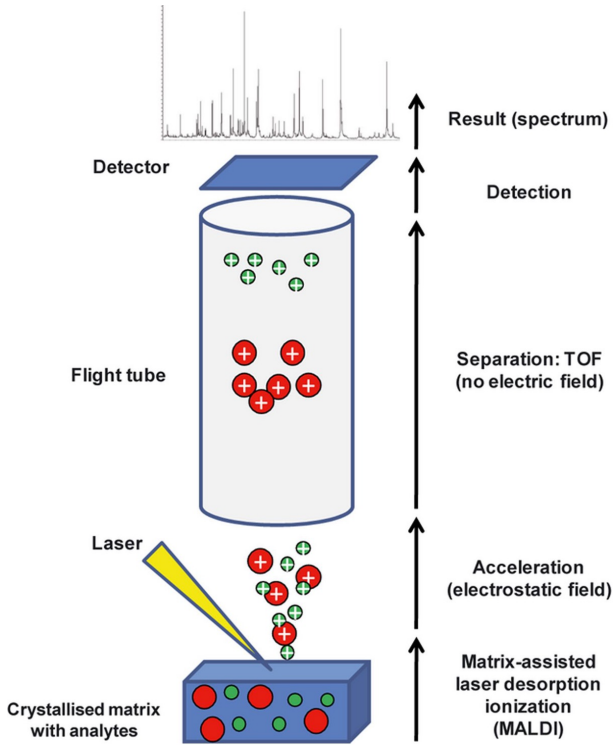


Figura 2.1: Descripción técnica de MALDI-TOF MS, extraído de [1].

En los últimos años, se ha extendido el uso de la espectrometría de masas MALDI-TOF como un método rápido y rentable para detectar ciertos mecanismos de resistencia antimicrobiana, diferenciando de forma veloz microorganismos como *Asperguillus fumigatus* [31], *Escherichia Coli* [19] y *Enterococcus Faecium* [32]. Estos estudios [19, 31, 32] demuestran que la integración de algoritmos de clasificación de ML basados en los espectros de masas extraídos con MALDI-TOF MS son idóneos para este propósito.

2.3. *Aspergillus fumigatus*

Uno de los hongos filamentosos patógenos más importante en humanos, *Aspergillus fumigatus*, es el causante de un conjunto de enfermedades entre las que se incluyen la aspergilosis invasiva, la aspergilosis broncopulmonar alérgica y la aspergilosis pulmonar crónica. Además este hongo es una de las principales causas infecciosas de muerte debido al aumento de pacientes con inmunodeficiencia [33] como puede ser los pacientes con el virus de la inmunodeficiencia humana (VIH) o aquellos que se han sometido a un trasplante de órganos.

El principal agente etiológico de la enfermedad aspergilosis es el *Aspergillus fumigatus Sensu Lato* y engloba diversas especies, siendo *A. fumigatus Sensu Stricto* la especie más frecuente y significativa en clínica. Además, las especies crípticas de este hongo representan el 10-15 % de los individuos, y están cobrando relevancia en el contexto de la resistencia a antifúngicos. Ambas especies suelen presentar resistencia a los azoles, sin embargo, las especies crípticas suelen mostrar también resistencia intrínseca a la anfotericina B [34, 35].

La resistencia a los azoles, tratamiento de elección para la aspergilosis, por los aislados de *Aspergillus fumigatus* es un problema manifestado alrededor del mundo debido a que los pacientes infectados por estas cepas resistentes, experimentan una evolución desfavorable. Se estima que los pacientes infectados por cepas con resistencia a los azoles presentan hasta un 31 % más de mortalidad en el día 42 que aquellos que son sensibles [34]. Además, es de importancia valorar que durante las últimas dos décadas en el norte de Europa, sobre todo en los Países Bajos y Reino Unido, se ha detectado un alto porcentaje de resistencia a este medicamento [34]. En consecuencia, es de gran importancia detectar esa resistencia y así tratar a los pacientes de aspergilosis de la forma más adecuada [36]. Para ello, su correcta identificación es esencial.

Los procedimientos moleculares suelen aplicarse en la diferenciación de las especies anteriores, pero necesitan métodos fenotípicos fiables para discriminar con precisión entre las especies crípticas de *Aspergillus*. La técnica MALDI-TOF MS se ha convertido en un procedimiento verdaderamente competente para la identificación rápida de diferentes especies de *Aspergillus* [31].

2.4. *Escherichia coli*

Escherichia coli, también conocido como *E. Coli*, es un género de la familia bacteriana de las *Enterobacteriaceae*. Es la parte de la microbiota más frecuente en el tracto gastrointestinal de humanos y animales de sangre caliente, y uno de los patógenos más importantes [37]. Debido a su función de comensal, vive en armonía con sus huéspedes sin causar enfermedades con facilidad. Sin embargo, en algunas ocasiones provoca graves patologías como diarrea [38], enteritis, infecciones urinarias, enfermedades respiratorias e infecciones del torrente sanguíneo. Además, algunas cepas se relacionan con lesiones de la enfermedad de Crohn [37].

En los últimos años se ha detectado una progresiva disminución en la sensibilidad de este microorganismo a los antibióticos que se utilizan para tratar las enfermedades que causa [39]. Además, se ha observado que *E. Coli* puede adquirir genes de resistencias de otras bacterias y transmitir sus propios genes a otros microorganismos del mismo tipo, lo que aumenta la preocupación mundial debido a la RAM en este caso [40]. Algunos de los antibióticos más utilizados para tratar las enfermedades por *E. Coli*, y que, por lo tanto, se estudia la respuesta de las bacterias ante ellos, son Ciprofloxacin, Ceftriaxone, Cefepime, Piperacillin-tazobactam y Tobramycin [19].

E. Coli resulta ser uno de los organismos más frecuentes en los laboratorios de microbiología clínica [29], siendo bastante habitual el uso de la técnica MALDI-TOF MS en su identificación distinguiéndola de las especies *Shigella* [41]. Además, esta método de identificación se utiliza para la diferenciación de poblaciones de la especie *E. Coli* que provocan diferentes patologías, puesto que causan gran cantidad de enfermedades en humanos [42].

2.5. Clover MS Data Analysis Software

CLOVER Bioanalytical Software S.L.U. es una empresa de bioinformática especializada en el tratamiento de espectrometría de masas y espectroscopia infrarroja (IR del inglés *Infrared Spectrometry*) para la identificación y discriminación de cepas microbianas así como otras aplicaciones de diagnóstico clínico utilizando herramientas de ML y de análisis estadístico. Se compone de un grupo de doctores y expertos con una amplia experiencia en investigación y el campo biomédico [20].

Clover MS Analysis Software Analysis (Clover MSDAS) es una plataforma online desarrollada por Clover Biosoft, la cual contiene herramientas de ML y análisis estadístico [2]. Esta

plataforma online es utilizada por profesionales de diversos equipos de investigación para tratar datos procedentes de espectros obtenidos por tecnologías MALDI-TOF MS, explicada en la Sección 2.2 y por Espectroscopía Infrarroja por Transformada de Fourier (FTIR). Este método de espectrometría FTIR tiene un amplio campo de aplicación, desde el análisis de pequeñas moléculas y complejos moleculares, hasta el análisis de células y tejidos, incluyendo el estudio de las proteínas analizando su conformación, su plegamiento y los detalles moleculares [43]. En combinación con ML, el análisis de los espectros FTIR puede emplearse en tareas como la distinción entre células sanas y patológicas [44] o la detección de bacterias aisladas [45], entre otras.

La mencionada plataforma permite realizar estudios de clasificación de microorganismos resistentes o sensibles a ciertos antimicrobianos, sin conocimientos de programación, mediante algoritmos de ML supervisados como RF, LightGBM , KNN (del inglés *k-nearest neighbors algorithm*), SVM (del inglés *Support Vector Machine*) o PLS-DA (del inglés *Partial least squares-discriminant analysis*) o no supervisados, como PCA (*Principal Component Analysis*).

En este TFG, se colabora con la empresa con el fin de que el algoritmo de BNN creado pueda implementarse en su plataforma. Además, los conjuntos de datos utilizados son proporcionados por Clover y su plataforma es utilizada para potenciar el estudio. Facilita la comparación de los resultados obtenidos de la BNN con otros algoritmos de ML supervisados que incluye, permite la visualización de los espectros de masas de las muestras y la detección de valores atípicos en los conjuntos de datos.

Capítulo 3

Métodos

En este capítulo se introducen los conceptos fundamentales relacionados con ML y el preprocesado de los espectros de masas previo al diseño de los modelos de ML. A continuación, se presentan y describen los métodos de ML usados en el desarrollo del presente TFG para el proceso de clasificación de microorganismos resistentes a antimicrobianos específicos, y finalmente, se exponen las prestaciones a calcular necesarias para la evaluación del rendimiento los modelos.

3.1. Introducción. Conceptos previos

A lo largo de la historia de la humanidad, las personas han utilizado herramientas y tecnologías para simplificar las tareas a realizar. En este marco, el ML destaca como herramienta revolucionaria definiéndose, según Arthut Samuel, como el campo de estudio que da a los ordenadores la capacidad de aprender sin ser programados explícitamente [46]. Tiene su origen en la década de 1950 con el movimiento de Inteligencia Artificial (IA) enfocándose en objetivos y aplicaciones prácticas, principalmente en la predicción y la optimización, aprendiendo y mejorando su rendimiento a través de la experiencia [47].

El ML, por tanto, se encuentra en la intersección entre las ciencias de la ingeniería, la computación y la estadística, demostrando ser una herramienta de gran utilidad para un número creciente de disciplinas [48]. En el ámbito del diagnóstico y la investigación en salud, los métodos de ML se utilizan beneficiándose de aquellos datos obtenidos de sensores y dispositivos médicos para evaluar el estado de salud del paciente. Estos modelos permiten descubrir asociaciones lineales y no lineales, interacciones y subgrupos que no pueden detectarse con

facilidad utilizando métodos convencionales [49]. Por ello, con la ayuda de estas técnicas, se puede aprender, analizar y extrapolar información relevante en la investigación médica, identificando condiciones y señales de alerta que podrían mejorar tratamientos y diagnósticos en medicina.

Los algoritmos empleados en un proceso de ML toman una matriz $X_{n\times m}$ compuesta de n muestras y m características. Cada elemento $X_{i,j}$ de la matriz representa el valor de la i-ésima instancia en la j-ésima variable. De esta manera, cada una de las muestras tiene asociada un vector de m características que las define. En algunas ocasiones, el conjunto de pares puede acompañarse por un vector $\mathbf{y}$ de longitud n cuyos elementos y_i corresponden a la etiqueta o valor asociado con la muestra correspondiente de la matriz X.

En líneas generales, se distinguen dos estrategias diferentes en el ML, aprendizaje supervisado y aprendizaje no supervisado, que se seleccionarán dependiendo de la tarea a abordar y del conjunto de muestras de las que se disponga.

- El **aprendizaje supervisado** trata de relacionar los vectores de características con las etiquetas propias de cada vector, conociéndose con anterioridad las etiquetas de cada uno de los vectores [47]. Para ello, se debe encontrar el modelo que relacione la entrada (matriz $X_{n\times m}$) con la salida (etiqueta). Los algoritmos de aprendizaje supervisado pueden dividirse a su vez en dos tipos:

  - **Clasificación**: el conjunto de etiquetas es numerable, prediciendo resultados categóricos.
  - **Regresión**: el conjunto de etiquetas no es numerable, siendo entonces los resultados de la predicción continuos.

  Algunas técnicas habituales de aprendizaje supervisados son la regresión lineal y no lineal, redes neuronales artificiales o los árboles de decisión, entre otros [47].

- En el **aprendizaje no supervisado**, sin embargo, el modelo intenta identificar relaciones y agrupaciones naturales entre los datos, pero sin hacer uso de la etiqueta asociada a cada observación, por lo que únicamente empleará el vector de características. Algunos ejemplos de estos algoritmos son k-medias y agrupamiento jerárquico (aglomerativo y divisivo [47]).

En el presente TFG se tratan modelos de aprendizaje supervisado, pues se conoce previamente la etiqueta asociada a cada una de las muestras que componen los conjuntos de datos

proporcionados por la empresa Clover. Principalmente, el estudio trata de predecir si ciertos microorganismos son sensibles o resistentes, tratándose entonces de una clasificación binaria (0 ó 1). Para llevar a cabo el propósito, se desarrollan diferentes algoritmos como BNN, RF y LightGBM, explicados en las Secciones 3.2.2, 3.2.4 y 3.2.5 respectivamente.

Los conjuntos de datos utilizados para el entrenamiento de los modelos de aprendizaje supervisado son obtenidos mediante instrumentos de espectrometría de masas MALDI-TOF a diversas muestras de distintos microorganismos y su posterior preprocesado. Este preprocesado se lleva a cabo en la plataforma de Clover y en él, existen diversas fases que se explican en detalle en la Subsección 3.1.1 como la estabilización de la varianza, técnicas de suavizado como*Saviztky-Golay*, corrección de la linea base con filtros como *Top-hat*, el proceso de filtrado de masas y la creación de matrices de picos con aquellos más relevantes de cada espectro.

3.1.1. Preprocesado

En la Sección 2.2 se define el resultado de la técnica MALDI-TOF MS como la representación del espectro de masas para cada muestra en la que se manifiesta la proporción entre masa y carga *m/z* en el eje de abscisas frente a su intensidad en el eje de ordenadas. El preprocesado de estos espectros es una etapa imprescindible para poder trabajar con ellos de manera uniforme, substrayendo el ruido de fondo o las variaciones que puedan obstaculizar su análisis posterior, dificultando así la obtención de conclusiones biológicas significativas. En esta etapa, los datos brutos que contienen valores *m/z* de analitos y derivados de varias formas de ruido, se convierten en datos con valores de *m/z* e intensidad más precisos [50]. El preprocesado de los espectros de masas puede dividirse en dos fases: la transformación de los espectros de masas y la búsqueda de los picos [51].

Transformación de los espectros de masas

Los valores de las intensidades de los espectros de masa pueden variar entre distintas mediciones para la misma muestra debido a las características propias de la técnica [51]. Por ello, esta fase trata de transformar de forma global y local los espectros mediante las siguientes técnicas:

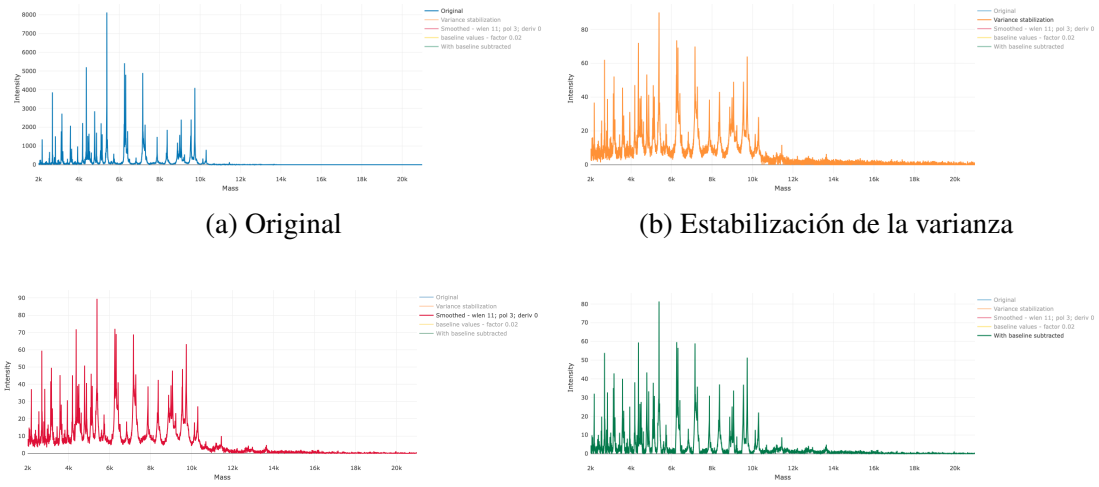
- La **estabilización de la varianza** consiste en utilizar una transformación de las intensidades de los espectros para obtener una mejor representación gráfica y evitar la dependencia entre la varianza y la media. Se pueden utilizar transformaciones como la de raíz cuadrada o la logarítmica [51, 52].

- Las técnicas de **suavizado** tratan de suavizar los espectros irregulares para facilitar la detección de los valores *m/z* de interés frente a los valores de ruido, mejorando la relación señal/ruido (SNR) [50]. Los métodos más simples se basan en utilizar una ventana que se desliza por el espectro ajustando la intensidad de cada valor del eje de abscisas en función de la intensidad de los valores vecinos. Destacan los algoritmos *Moving Average filter, Savitzky-Golay filter, Gaussian filter* y *Kaiser window* entre otros [53].

- Las perturbaciones químicas pueden provocar un desplazamiento de las intensidades de los picos dependiente del valor de *m/z*, conocido como línea base. La **corrección de la línea base** trata de controlar la amplificación de este ruido químico. El método *Top-hat* se basa en un filtro no lineal que trata de sustraer la apertura morfológica del espectro original eliminando los objetos más grandes [54]. Suele eliminar las tendencias lentas mejorando el contraste.

- El proceso de **filtrado de masas** establece un intervalo de masas de interés en el que se concentre la información relevante del espectro de masas, desechando aquellas que no pertenezcan a él. Un ejemplo sería el intervalo [2000, 20000] *m/z*, utilizado frecuentemente por Clover.

En la Figura 3.1 se representa el ejemplo de la transformación efectuada en un espectro de masas: estabilización de la varianza, un suavizado de ruido con el filtro *Savitzky-Golay* con una longitud de ventana de 11 puntos basándose en una regresión polinomial de grado 3 y corrección de la línea base.

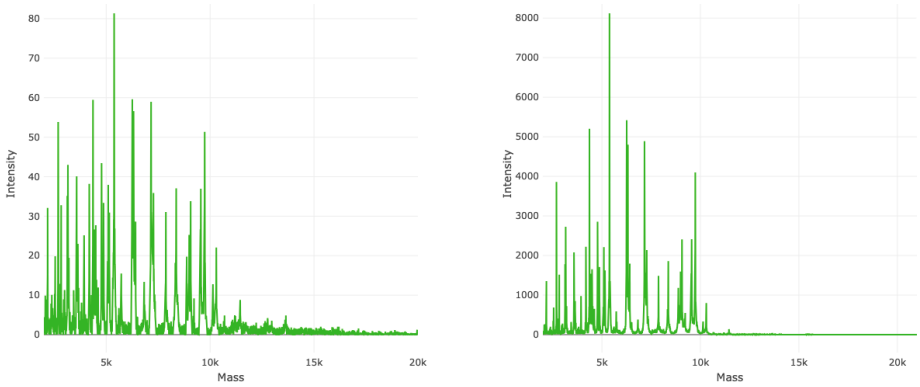


(a) Original (b) Estabilización de la varianza

(c) Suavizado (d) Corrección de la línea base

Figura 3.1: Etapa de transformación de los espectros de masas, extraído de [2].

A continuación, en la Figura 3.2 se muestra la comparativa del espectro de masas original frente al obtenido al final de esta etapa tras haber filtrado los valores de masas en un intervalo comprendido entre 2000 y 20000 *m/z*.



(a) Original (b) Final

Figura 3.2: Espectros de masas original (a) y resultante tras la fase de transformación (b), extraído de [2].

Estas técnicas actúan individualmente en cada uno de los espectros de masas, no influyendo en el conjunto, por lo que se puede aplicar tanto al conjunto de datos por completo, como por separado en los subconjuntos *train* y *test*, detallados en la Sección 3.1.2.

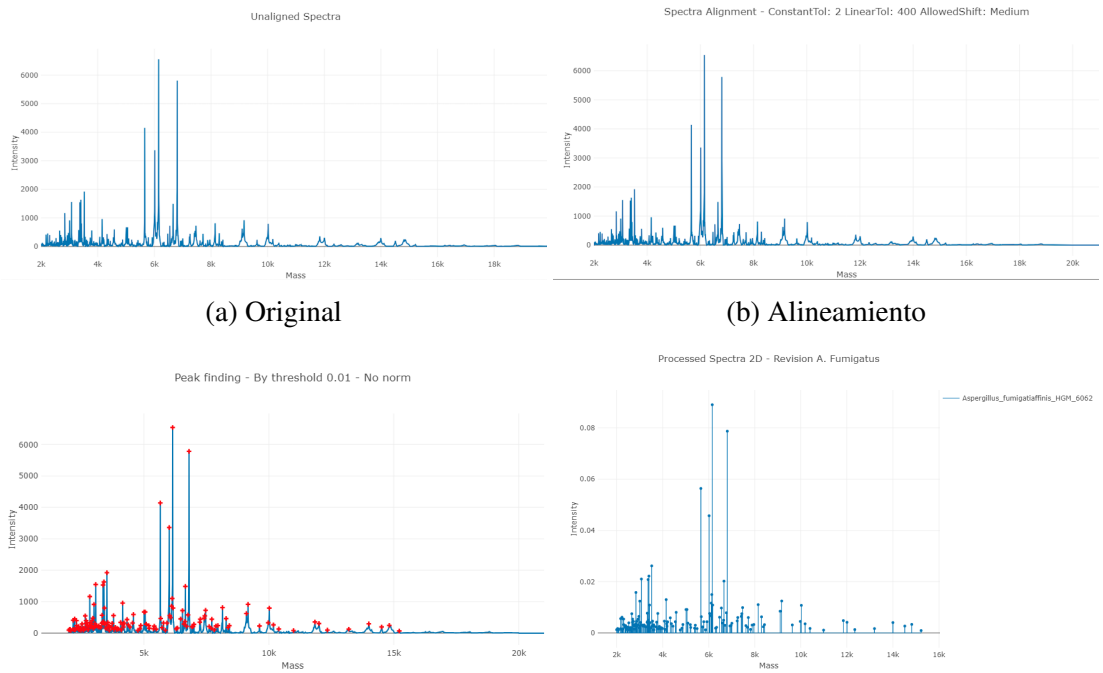
Búsqueda de picos

La búsqueda de picos es un paso de gran importancia ya que son los que permiten identificar los diferentes compuestos de la muestra. En los estudios de predicción de microorganismos sensibles o resistentes, la información relativa a los picos constituye el vector de características de cada una de las muestras tomado por el algoritmo de ML para la clasificación de las observaciones. Para ello, se suelen seguir los siguientes pasos:

- El proceso de **alineamiento** trata de corregir el desplazamiento en la posición de los picos del espectro debido a múltiples factores como variaciones en el tiempo de vuelo o la calibración del sistema. Para ello se requieren de algoritmos que para cada espectro, encuentren una función de curvatura con el fin de sincronizar todos los espectros [55].
- La mayoría de métodos de **detección de picos** tratan de descartar los valores de *m/z* de intensidad menor a un umbral establecido. Además, se establecen una tolerancia, en la que dentro de ella, dos picos desplazados se consideran el mismo pico.

- La técnica de **normalización** trata de preservar la proporcionalidad entre la intensidad de los picos. Para ello, el método más común es el de Corriente Iónica Total o TIC, que se basa en el hecho de que la intensidad total de la señal en un espectro está relacionada con la cantidad total de iones de la propia muestra [56].

En la Figura 3.3 se expone un espectro que ha sido tratado por las técnicas de la fase anterior. A continuación este espectro ha sido alineado, se han detectado los picos y se ha llevado a cabo una normalización de la intensidad, pasando de un rango entre 0 y 8000, a un rango entre 0 y 0,08.



(a) Original

(b) Alineamiento

(c) Detección de picos

(d) Normalización

Figura 3.3: Etapa de búsqueda de picos, extraído de [2].

Tras la realización de estos métodos, se obtiene una matriz de picos (*peak matrix* en inglés) en la que se ha reducido la dimensionalidad con respecto a la información inicial y será el conjunto de datos de entrada para los algoritmos de ML. En este contexto, la citada matriz es una caracterización de los picos de la espectrometría de masas MALDI-TOF que contiene la información normalizada sobre la intensidad de los picos del espectro de cada muestra para cada valor *m/z*.

Sin embargo, en este caso sí es importante aplicar las técnicas en los subconjuntos *train* y *test* por separado, de manera que la matriz de picos de *test* se obtenga en función de la resultante

en *train* para que ambos conjuntos tengan la misma cantidad de picos y por ende, la misma cantidad de variables que definan cada una de las muestras.

En este estudio se llevan a cabo ambas fases de la etapa de preprocesado en la plataforma de Clover Biosoft.

3.1.2. Etapas de diseño

Una vez seleccionado el algoritmo más adecuado en relación con el problema a abordar, se lleva a cabo el proceso de diseño del modelo mediante la realización de dos etapas: fase de entrenamiento y fase de test. Éstas son imprescindibles para que el aprendizaje del modelo sea adecuado y se lleve a cabo una correcta evaluación de su capacidad de generalización, evitando el sobreajuste [57]. Para la creación de estas fases, el conjunto de datos se divide en dos grupos independientes: subconjunto de *train* y el subconjunto de *test*. Generalmente la división se hace en una proporción del 70 % y 30 % respectivamente, o del 80 % y 20 %, aunque otras posibilidades son válidas.

- **Fase de entrenamiento**: en esta etapa, el modelo se entrena utilizando el subconjunto de *train*. Además, en esta fase se ajustan los hiperparámetros del modelo de forma manual o mediante la técnica de validación cruzada (CV, del ingles *cross-validation* [58]. Los hiperparámetros son coeficientes que se optimizan o aprenden automáticamente en esta fase, son distintos a los parámetros del modelo [59].

  En este trabajo, se ha utilizado validación cruzada de *k* particiones (*K-Fold Cross-Validation*) en la que el conjunto de entrenamiento se divide en k subconjuntos, también llamados *folds*, formados por la misma cantidad de muestras. De estas particiones, k-1 se utilizan para entrenar el modelo, mientras que el conjunto restante se emplea para predecir y medir la eficacia del modelo. Este proceso se lleva a cabo k veces, modificando el subconjunto que se emplea para validar, de manera que en cada una de las k particiones, el subconjunto de validación sea diferente (véase la Figura 3.4) [60]. Concretamente, ha sido aplicada la (*Statrified K-Fold Cross-Validation*), que se diferencia de la (*K-Fold Cross-Validation*) en que en la primera se asegura que la distribución de las clases sea representativa en cada uno de los *folds*, lo que ayuda a lidiar con el problema del desbalanceo de clases que se detalla en la Sección 4.1.

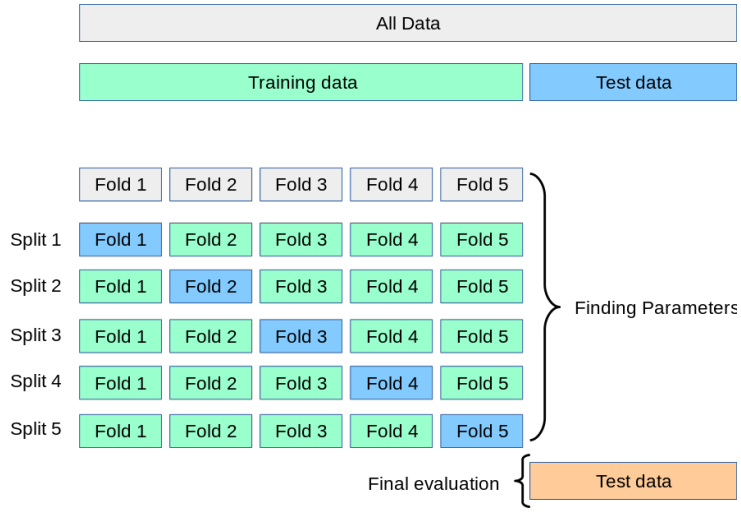


Figura 3.4: Ejemplo de *5-fold cross-validation*, extraída de [3].

En este caso, se toman 5 particiones, k = 5, de manera que en cada una de las particiones, el 80 % de las muestras se utiliza para entrenar el modelo y el 20 % restante para validarlo. Basándose en esta técnica de CV, se determinan los valores de los hiperparámetros que mejor generalicen el modelo [58], empleando el método *GridSearchCV* (aportado por la librería Sklearn de Python) [60]. Para ello, es necesario seleccionar una figura de mérito, evitando el sobreajuste (del ingles, *overfitting*), tal y como se muestra en la Figura 3.5.

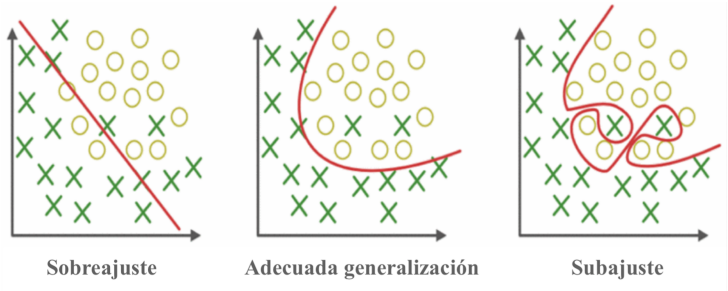


Figura 3.5: Ejemplo de generalización de un modelo a las muestras, adaptada de [4].

- **Fase de evaluación**: en esta etapa el modelo creado anteriormente se evalúa con el subconjunto de *test*, midiendo el desempeño del modelo creado a partir de varias figuras de mérito [58].

En el presente TFG, para el diseño de los modelos seleccionados (BNN, RF y LightGBM) se ha procedido a hacer una división con un porcentaje del 70 % en entrenamiento y 30 % en evaluación, simplemente por ser uno de los porcentajes más utilizados [61]. Con el subconjunto *train* se entrenan los modelos y se seleccionan los hiperparámetros a partir de la técnica

Statrified 5-fold cross-validation, utilizado como figura de mérito *balanced accuracy*, que se explicará en la Sección 3.3. Asimismo, con el subconjunto *test* se evalúan las predicciones de los modelos adquiriendo las figuras de mérito.

3.1.3. Normalización de características

La normalización de características es un método mediante el que se transforma las variables a una escala comparable, garantizando que los rangos valores de las variable sean semejantes [62]. Existen diversos métodos pero los más comunes son la normalización min-max y la normalización z-score.

La primera técnica, la normalización min-max, trata de normalizar los valores de las características para que estén en un rango específico definido por límites inferior y superior considerando la siguiente ecuación: $\tilde{X}^{(n)}_{train,j} = \frac{X^{(n)}_{train,j} - X_{train,jMIN}}{X_{train,jMAX} - X_{train,jMIN}}$, $\tilde{X}^{(n)}_{test,j} = \frac{X^{(n)}_{test,j} - X_{train,jMIN}}{X_{train,jMAX} - X_{train,jMIN}}$

La normalización z-score transforma los valores de las variables con el objetivo de que la media, μ de los valores sea nula y la desviación típica σ tenga valor unidad. Se calcula mediante la siguiente ecuación: $\tilde{X}^{(n)}_{\text{train},j} = \frac{X^{(n)}_{\text{train},j} - \mu_{\text{train},j}}{\sigma_{\text{train},j}}$, $\tilde{X}^{(n)}_{\text{test},j} = \frac{X^{(n)}_{\text{test},j} - \mu_{\text{train},j}}{\sigma_{\text{train},j}}$

En ambas ocasiones, para la normalización de cada característica j en los conjuntos, se utilizan los valores correspondientes de *train* (valores máximo y mínimo en el primer método, y valor de la media y la desviación típica en el segundo) para que el proceso de normalización se lleve a cabo en función de este conjunto. Este proceso, por tanto, tiene lugar previamente al entrenamiento del modelo de aprendizaje supervisado, tanto en el subconjunto de datos de *train* como en el de *test*.

En el presente TFG no se realiza una normalización de los datos utilizando estos métodos pues, como se refiere en la Sección 3.1.1, en el preprocesado de los espectros de masa se realiza una normalización previa a la definición final de la matriz de picos mediante el método TIC.

3.1.4. Análisis de Componentes Principales

El Análisis de Componentes Principales es una técnica de análisis de datos utilizada en este contexto para reducir la dimensionalidad de un conjunto de variables. Para ello, se calculan los vectores propios de la matriz de covarianza de las características originales, obteniéndose un conjunto menor de variables no redundantes considerando aquellos componentes principales que explican una gran fracción de la variabilidad total de los datos [63]. La técnica PCA destaca

la contribución de las distintas características a través de sus componentes principales [64], que se visualiza mediante el cálculo de la varianza acumulativa de cada una de las variables. Con estos valores, se toman tales cuya variabilidad es significante, en este proyecto se toman aquellas que tienen entre un 80 y un 85 %, permitiendo despreciar las menos relevantes.

3.2. Métodos de Aprendizaje Automático

Los métodos de ML permiten diseñar un algoritmo capaz de descubrir asociaciones no lineales, interacciones y subgrupos entre un conjunto de observaciones que tienen un serie de características. A continuación se explican los métodos de ML supervisados que se desarrollan en la clasificación de dos conjuntos de muestras distintos, descritos en la Sección 4.1.

3.2.1. Red Neuronal Artificial

Las Redes Neuronales Artificiales (ANN, del inglés *Artificial Neural Network*) son métodos de ML basados en sistemas computacionales que se componen una gran cantidad de procesadores simples conectados entre sí [65]. Se inspiran en las redes neuronales que forman el sistema nervioso humano y se componen de dendritas, soma y axón (véase en la Figura 3.6). Las dendritas son las ramificaciones externas que captan los impulsos nerviosos que emiten otras neuronas. El soma o cuerpo se encarga de procesar esos impulsos que llegan a la neurona, y el axón se encarga de transmitir la información procesada emitiendo un impulso nervioso.

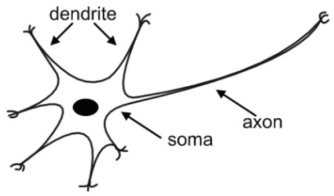


Figura 3.6: Estructura de una neurona biológica, extraída de [5].

En las ANN, las neuronas artificiales corresponden con las neuronas biológicas, siendo los nodos de la red. La Figura 3.7 representa la estructura esquemática de una neurona artificial. Cada una de las entradas que llegan a ellas, se multiplican por unos valores llamados pesos y se suman entre sí, correspondiendo al impulso nervioso que capta la neurona. El resultado de este sumatorio, es procesado por la neurona mediante una función de activación obteniendo un valor de salida de la neurona, el impulso nervioso de salida.

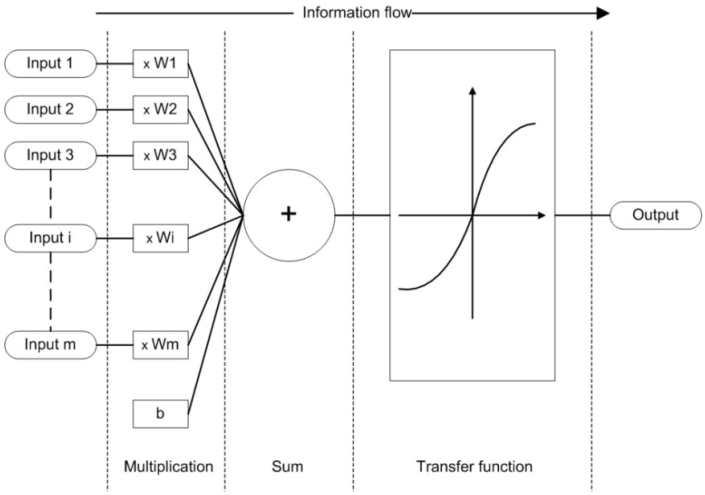


Figura 3.7: Estructura detallada de una neurona artificial, extraída de [5].

Las neuronas se agrupan en diversas capas que constituyen las etapas de la red. Aquellas que forman la capa de entrada reciben los datos reales que se pretende tratar con el modelo, existiendo un nodo por cada característica de las muestras. A continuación, se encuentran las capas ocultas (*hidden layers* en inglés) que determinan la complejidad de la red y conectan con la capa de salida, formada por tantas neuronas como salidas diferentes experimente el modelo. En la Figura 3.8 se representa la estructura de una ANN del tipo Perceptrón Multicapa que se compone de una única capa oculta.

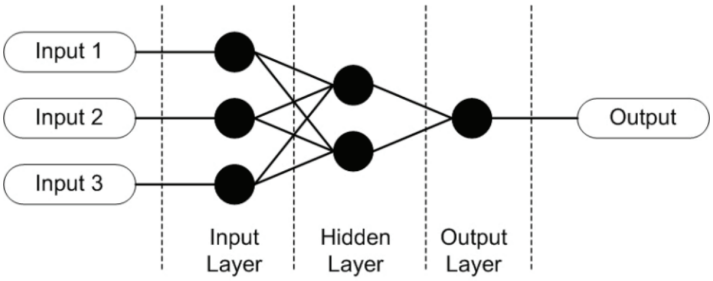


Figura 3.8: Red Perceptrón Multicapa, extraído de [5].

El proceso de aprendizaje de las ANN se lleva a cabo con la técnica de retropropagación (*backpropagation* en inglés), que permite el aprendizaje supervisado de la red mediante la obtención de valores óptimos para los pesos presentes en las interconexiones neuronales [66]. Para llevar a cabo este procedimiento, se definen un número de máximo de épocas (*max epochs* en inglés) y el tamaño del *batch* (*batch size* en inglés). Las épocas representan la cantidad máxima

de ciclos que lleva a cabo el algoritmo y el tamaño del *batch* define la cantidad de muestras que se utiliza en cada uno de los ciclos para estimar el error del modelo.

Inicialmente, el algoritmo *backpropagation* asigna de forma aleatoria los pesos. A continuación, la capa de entrada recibe el vector de las muestras de entrenamiento y se calcula la salida con los pesos atribuidos, comparándola con la esperada. Si los valores coinciden, los pesos se mantienen; sin embargo, si las salidas son diferentes, los pesos deben ajustarse optimizando la función de coste, que cuantifica la discrepancia entre la salida deseada y la obtenida por la red. Este proceso se repite hasta que la función de coste sea inferior a un umbral establecido o hasta que se realicen las épocas definidas [67].

Cabe destacar el impacto que tiene la forma de conexión entre neuronas en el funcionamiento de la red, diferenciando dos categorías [5]:

- *Feed-Forward*: esta arquitectura es lineal, no existe una conexión entre la salida de una capa con las neuronas anteriores.
- *Feedback*: en este tipo de conexión, la salida de una capa se convierte en la entrada de una capa anterior o de ella misma.

3.2.2. Red Neuronal Bayesiana

Una Red Neuronal Bayesiana (BNN, del inglés *Bayesian Neural Network*) es una adaptación de la ANN que incorpora conceptos de inferencia bayesiana para el entrenamiento del modelo mejorando su rendimiento [68].

La inferencia bayesiana define un modelo de probabilidad que incorpora algún tipo de conocimiento previo acerca de un parámetro [69]. Este modelo combina los parámetros previos y la probabilidad para obtener la probabilidad posterior siguiendo la siguiente ecuación, que se basa en el Teorema de Bayes [70]:

$$P(\theta|y) = \frac{P(y|\theta) \cdot P(\theta)}{P(y)}$$

Donde $P(\theta|y)$ es la probabilidad de que se cumpla θ si sucede y, conocida como probabilidad posterior; $P(y|\theta)$ es la probabilidad de que se cumpla y si θ es cierto, conocido como la función de verosimilitud de las observaciones, $P(\theta)$ es el *prior* (distribución de probabilidad que representa el conocimiento previo) y $P(y)$ es la evidencia.

El objetivo del modelo BNN es explotar los beneficios de la modelización probabilística y de la capacidad de las redes neuronales para ser aproximadores universales de funciones [70], más robustos y con mejor generalización que las ANN estándar [71]. Para ello, en lugar de ser valores únicos (véase en la Sección 3.2.1), los pesos de la BNN se entrenan como distribuciones probabilísticas [68, 70, 71]. En consecuencia, las BNN son más difíciles de entrenar y requieran mayor cantidad de datos de entrada y tiempo de procesamiento. Además, requieren de conocimientos probabilísticos y estadísticos.

Existen diversos tipos de *priors* en función de la distribución que se seleccione. En este TFG, el *prior* utilizado es el Gaussiano que suele ser el más usado por su simplicidad y atractivo computacional [72]. El *prior* Gaussiano representa una distribución normal con una media μ y desviación típica σ.

En el presente TFG, la BNN sigue una tarea de clasificación y al ser una ANN, su proceso de aprendizaje se efectúa con la técnica *backpropagation*, definiendo una función de coste según la pérdida de entropía cruzada (*cross entropy loss* en inglés) [73] y la pérdida binaria de Kullback-Leiber (BKLLoss, del inglés *Binary Kullback-Leibler Loss*) [74]. Además, se utiliza el optimizador Adam, que es uno de los algoritmos de optimización de gradiente descendente más populares en ANN [75]. Asimismo, la arquitectura de la BNN es lineal, siguiendo una forma de conexión *Feed-Forward*.

En el diseño de la BNN se requiere un proceso de optimización de hiperparámetros en el que se toman los siguientes:

1. ***Batch size***: cantidad de datos que tiene cada iteración de un ciclo.
2. ***Max epoch***: número máximo de veces que se ejecuta el algoritmo *backpropagation*.
3. ***Activation Function***: función matemática que modifica el valor resultante a la salida de la neurona pudiendo introducir no linealidad. Existen diversos tipos como la función ReLu (del inglés *Rectified Lineal Unit*) que transforma los valores anulando los negativos, la función Sigmoid que transforma los valores al rango [0, 1] o la función Softmax que convierta las salidas de la capa anterior en probabilidades que suman 1.
4. ***Hidden neurons***: Número de neuronas que forman la capa oculta.

3.2.3. Árboles de Decisión

Un árbol de decisión es un modelo de aprendizaje supervisado no lineal y no paramétrico que en función del tipo de salida, será un regresor o un clasificador [76]. En este caso, nos centramos en la técnica de clasificación (véase en la Figura 3.9 un ejemplo representativo de este modelo).

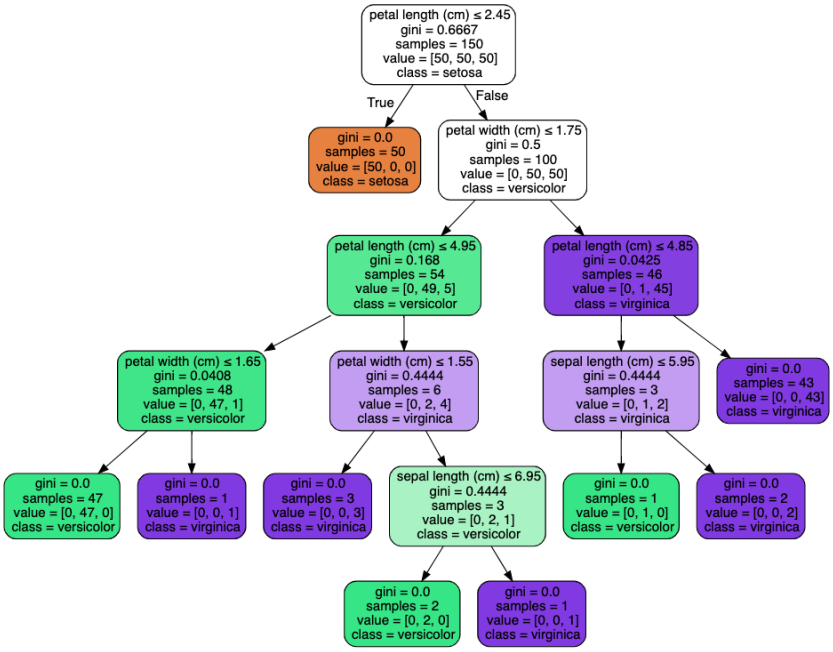


Figura 3.9: Ejemplo de clasificación por árbol de decisión, extraído de [6].

Como representa la Figura 3.9, un árbol de decisión comienza con un nodo raíz, que no tiene ramas entrantes. Las ramas salientes del nodo raíz alimentan los nodos internos, también conocidos como nodos de decisión. En función de las características disponibles, ambos tipos de nodos realizan evaluaciones para formar subconjuntos homogéneos, que se indican mediante nodos hoja o nodos terminales. Los nodos hoja representan todos los resultados posibles dentro del conjunto de datos [76].

En este modelo, los hiperparámetros que suelen optimizarse son la profundidad máxima y el número mínimo de muestras. La **profundidad máxima** es el valor entero que representa la longitud máxima del camino desde el nodo raíz hasta una hoja. Cuanto menor sea este valor, menor será el tamaño del árbol y la tasa de acierto en la fase de *train*, obteniendo una buena generalización. El **número mínimo de muestras** es la cantidad mínima de instancias que pertenecen a la misma clase en cada nodo hoja.

3.2.4. Clasificador Random Forest

El modelo Bosque Aleatorio (RF, del inglés *Random Forest*) es una técnica de aprendizaje automático supervisada que se basa en árboles de decisión [77]. Es un algoritmo por conjuntos utilizado con éxito en grandes bases de datos ya que puede manejar miles de variables de entrada sin un proceso de selección de características, formando una estimación no sesgada del error de generalización. Es robusto a valores atípicos y al ruido, y es uno de los mejores métodos basados en conjuntos de árboles desde una visión computacional [77].

En RF, los árboles de decisión se combinan con la técnica *bagging* [78], de manera que los distintos árboles que forman el modelo utilizan diferentes porciones de datos de entrenamiento en la fase de *train* (véase en la figura 3.10). Cada uno de los árboles de decisión devolverá una predicción sobre la etiqueta de las muestras con el fin de exponer una predicción conjunta correcta del modelo. Ésta se define por el tipo de salida con más votos, es decir, la clase predicha es aquella definida por el mayor número de árboles de decisión individuales [78].

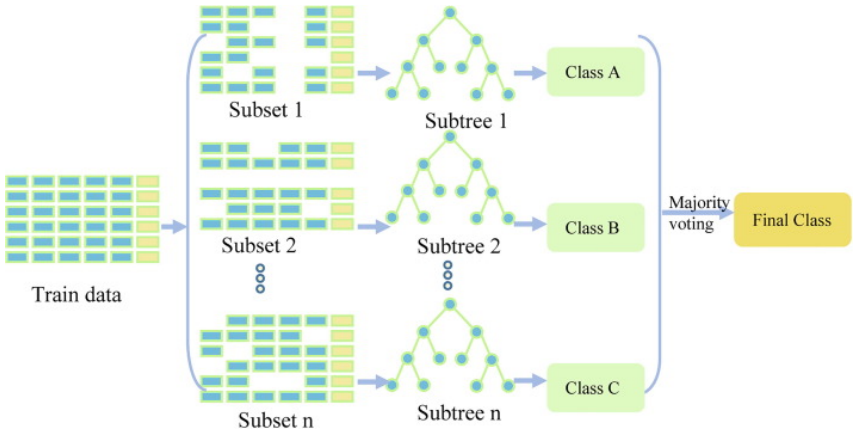


Figura 3.10: Técnica *bagging* aplicada a un clasificador Random Forest formada por 3 árboles, extraído de [7].

En el modelo RF, además de los hiperparámetros anteriores que se optimizan en los Árboles de decisión, también se seleccionan la cantidad de árboles, el número máximo de características a tener en cuenta para buscar la mejor división y el número mínimo de muestras para dividir el nodo.

3.2.5. Clasificador LightGBM

La Máquina de Reforzamiento de Gradiente Ligero, conocida como LightGBM (del inglés *Light Gradient Boosting Machine*), es un algoritmo de aprendizaje automático que supone una implementación eficiente de árboles de refuerzo de gradiente [7]. El algoritmo LightGBM modelo ha sido comparado con otras implementaciones y se ha observado que reduce el tiempo de entrenamiento hasta 20 veces gracias a técnicas novedosas que incluyen [79].

Se caracteriza por combinar el algoritmo de árbol de decisión basado en el muestreo unilateral por gradiente (GOSS, del inglés *Gradient-based One-Side-Sampling*), la agrupación exclusiva de características (EFB, del inglés *Exclusive Feautre bundling*) y una estrategia de crecimiento por histogramas y hojas con un límite de profundidad [7, 80]. GOSS selecciona las instancias más relevantes y EFM trata de reducir el número total de características agrupando aquellas que son mutuamente exclusivas [79].

Algunos de los hiperparámetros a optimizar en este caso son el número de árboles requerido, el número de hojas de cada árbol, su profundidad y la tasa de aprendizaje que minimiza el sobreajuste.

3.3. Figuras de mérito

Este Capítulo se concluye haciendo una presentación de las distintas figuras de mérito que evalúan el rendimiento de los clasificadores diseñados en el Capítulo 5.

Para la correcta explicación de las métricas, es conveniente considerar la matriz de confusión. Ésta es una tabla cruzada que contiene el número de ocurrencias de las diferentes clases de la clasificación real, y la manera en la que el modelo las categoriza (clasificación predicha). Las filas, por tanto, representan la clase real de las instancias y las columnas, su clase predicha. Las clases se enumeran en el mismo orden en las filas que en las columnas, por lo que las instancias que se han clasificado adecuadamente se localizan en la diagonal principal, desde arriba hacia abajo [81]. Las dimensiones de la matriz de confusión varían conforme si la clasificación es binaria (véase en la Tabla 3.1) o multiclase.

La clase 0 es considerada como la predicción negativa, y la clase 1, como la predicción positiva. Por lo tanto, si el algoritmo predice la etiqueta positiva y realmente es positiva, las predicciones se definen como verdaderos positivos (VP). Si el algoritmo predice la etiqueta positiva como negativa, éstas son falsos negativos (FN); si se predice la etiqueta negativa como positiva, falso positivo (FP); y si la etiqueta negativa se predice como negativa, verdadero negativo (VN).

		Valor Clasificación	
		Clase=1	Clase=0
Valor Real	Clase=1	Verdaderos positivos (VP)	Falsos negativos (FN)
	Clase=0	Falsos positivos (FP)	Verdaderos negativos (VN)

Tabla 3.1: Matriz de confusión para clasificación binaria.

A partir de los valores VP, FN, FP y VN, se pueden calcular las siguientes métricas para evaluar las prestaciones del modelo [81]:

Accuracy

La tasa de acierto (en inglés *accuracy*) indica la proporción de instancias que se han clasificado correctamente en relación con el número total de muestras del conjunto evaluado. Se define por la siguiente ecuación:

$$Acurracy = \frac{VP+VN}{VP+VN+FP+FN}$$

Precision

La precisión (en inglés *precision*) informa sobre el porcentaje de aciertos en las instancias clasificadas como positivas. Definido por la siguiente ecuación:

$$Precision = \frac{VP}{VP+FP}$$

Recall

La sensibilidad (en inglés *recall*) informa sobre la cantidad de instancias de la predicción positiva que se han clasificado correctamente. También se le conoce como Tasa de Verdaderos Positivos (TVP). Definido por la siguiente ecuación:

$$Recall = \frac{VP}{VP+FN}$$

Specificity

La especificidad (en inglés *specificity*) es la probabilidad de clasificar correctamente a los casos de la clase negativa con respecto al total de muestras realmente negativas. Definida por la siguiente ecuación:

$$Specificity = \frac{VN}{VN + FP}$$

A partir de esta figura de mérito se puede calcular la Tasa de Falsos Positivos (TFP) como $TFP = 1 - especificidad$.

Balanced Accuracy

La exactitud equilibrada (en inglés *balanced accuracy*) es la media aritmética entre la sensibilidad y la especificidad, que se utiliza para medir el rendimiento del modelo cuando las clases están desbalanceadas.

$$BalancedAccuracy = \frac{Recall + Specifity}{2}$$

ROC Curve

La curva estadística operativa del receptor (ROC, del inglés *Receiver Operating Characteristic*) representa la TVP (sensibilidad) frente a la TFP ($1 - especificidad$) para un sistema clasificador binario a medida que varía el umbral. El punto más a la izquierda de la curva corresponde a clasificar todas las muestras como la predicción negativa, y el punto más a la derecha, como la predicción positiva [48].

AUC

La curva ROC, citada anteriormente, es útil para evaluar modelos de clasificación. Sin embargo, puede ser más complicado emplearla en la comparación entre modelos. En este caso, se requiere del área bajo la curva (AUC, del inglés *Area Under the Curve*) puesto que proporciona el valor de la integral de la curva [82].

Capítulo 4

Bases de datos y análisis descriptivo

En este capítulo se presentan las bases de datos consideradas en el TFG, así como una descripción de las etapas de preprocesamiento seguidas y el análisis exploratorio de los datos.

4.1. Descripción de las bases de datos

En este estudio, se toman dos bases de datos que permitirán evaluar el rendimiento de la BNN creada. Éstas han sido proporcionadas por CLOVER Bioanalytical Software S.L.U. y son el resultado de utilizar la técnica MALDI-TOF MS para la identificación de microorganismos.

4.1.1. *Aspergillus fumigatus*

En primer lugar, se toma el conjunto de datos utilizado en [31] que contiene 175 muestras del hongo *Aspergillus fumigatus*. Por un lado, estos ejemplos pueden clasificarse en dos grupos diferentes: la especie *Aspergillus fumigatus sensu stricto (s.s.)* y las especies crípticas, habiendo 149 muestras de las primeras y 26 de las segundas, lo que supone una representación de la realidad ya que, tal y como se ha comentado en la Sección 2.3, las especies crípticas representan un 10-15 % de los individuos de este hongo.

Por otro lado, la especie *A. fumigatus s.s.* puede clasificarse en resistente y sensible a las azoles en ausencia de una exposición previa, siendo 43 de las 149 muestras de la especie *A. fumigatus s.s.* resistentes, y las 106 restantes sensibles. Cada una de las instancias se define por la intensidad del espectro en las diferentes masas espectrales, correspondiéndose con 2302 características.

En las Figuras 4.1, 4.2 y 4.3 se representan los espectros de las instancias de las tres posibles clases (especies crípticas, resistente y sensible) en este conjunto. En primer lugar, de color azul, se grafican las muestras pertenecientes a las especies crípticas con un rango entre 2000 *m/z* y 20.000 *m/z*. Las muestras de *A. fumigatus s.s.* sensible a las azoles se representan de color verde en un rango entre 2000 *m/z* y 25.000 *m/z* y las resistentes, de color rojo entre 2000 *m/z* y 20.000 *m/z*. Se observa una clara distinción entre los espectros de masas de las especies crípticas con respecto a las demás, puesto que la disposición de los picos de las dos categorías de la especie *A. fumigatus s.s.* (Figuras 4.2 y 4.3) es más semejante entre ellas. Asimismo, la representación de los espectros de masas de la categoría sensible expone picos con mayor intensidad que la categoría resistente.

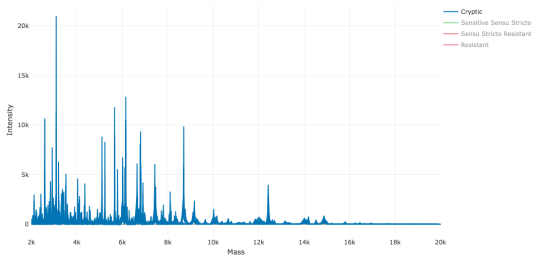


Figura 4.1: Espectros de masas de las especies crípticas, extraído de [2].

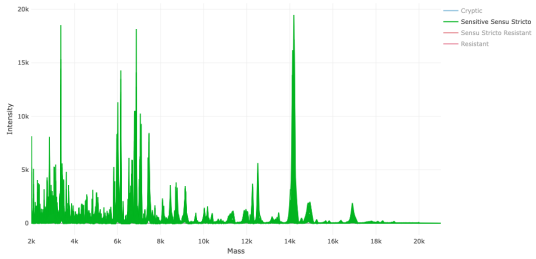


Figura 4.2: Espectros de masas de la categoría sensible, extraído de [2].

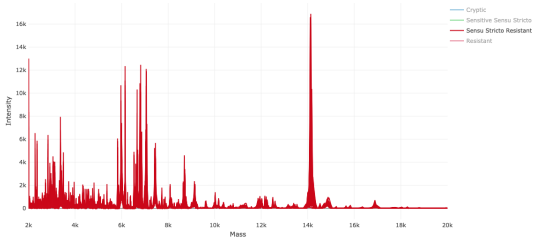


Figura 4.3: Espectros de masas de la categoría resistente, extraído de [2].

4.1.2. *Escherichia coli*

El segundo conjunto de datos contiene muestras de la bacteria *E. Coli*. Es una adaptación con menor cantidad de instancias que el utilizado en [19], reduciendo los 1394 ejemplos asociados al año 2018 a 296 ejemplos, con el objetivo de evaluar un algoritmo de ML basado en el espectro de masas MALDI-TOF para su futura implementación en la optimización del tratamiento y la administración de antibióticos, combatiendo la resistencia a estos antimicrobianos. Estas bacterias pueden ser resistentes, sensibles o presentar una situación intermedia a 82 antibióticos como Ciprofloxacin, Ceftriaxone o Cefepime. Se compone de 296 muestras de las que se disponen 6934 valores de masas espectrales en un intervalo de 2000 *m/z* a 20000 *m/z*, que definen los espectros de cada una de las instancias.

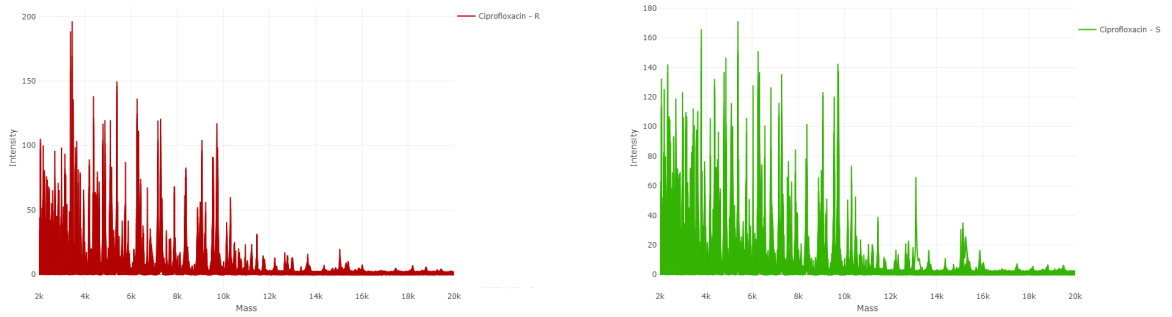
En este estudio, el principal objetivo es valorar el rendimiento del modelo de BNN creado, por lo que éste se ejecuta para los antibióticos Ciprofloxacin, Ceftriaxone o Cefepime, puesto que son tres de los más representativos según los resultados obtenidos en el estudio de referencia [19]. De las 296 muestras, como se refleja en la Tabla 4.1, 114 son resistentes al antibiótico Ciprofloxacin y 172 son sensibles; 110 son resistentes y 181 sensibles al Cefriaxonee; 50 son resistentes y 193 son sensibles al Cefepime.

Respuesta al antibiótico	**Ciprofloxacin**	**Ceftriaxone**	**Cefepime**
Resistentes	114	110	50
Sensibles	172	181	193
Total	286	191	243

Tabla 4.1: Distribución de las muestras de *E. Coli* según la respuesta que provocan ante los antibióticos Ciprofloxacin, Ceftriaxone y Cefepime.

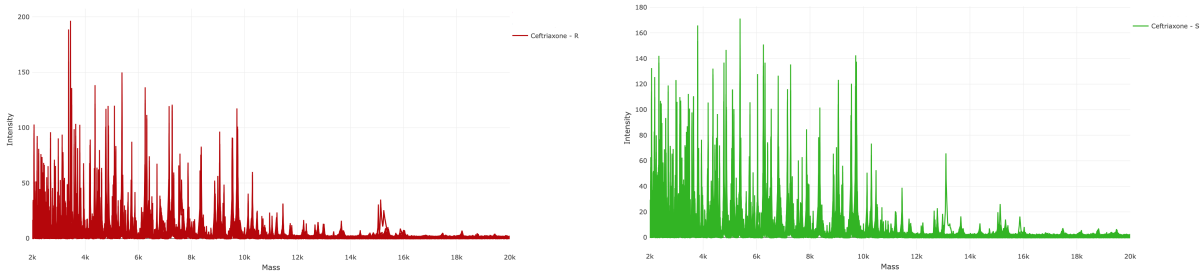
En la Figura 4.4a se representan las muestras resistentes al antibiótico Ciprofloxacin y en la Figura 4.4b, aquellas que son sensibles a este fármaco. Ambas grafican los espectros de masas representando la intensidad frente a los valores de masa *m/z*. En el caso de la Figura 4.5, se representan los espectros de masas de las muestras que son resistentes y sensibles, respectivamente, al Ceftriaxone. Finalmente, se grafican los espectros de masas de las muestras resistentes y sensibles al Cefepime en la Figura 4.6.

Atendiendo a las Figuras 4.4, 4.5 y 4.6, los espectros de masas de las muestras sensibles a los tres antibióticos presentan valores de las intensidades de los picos superiores a los de las resistentes. Además, se observa un mayor volumen en los espectros de los ejemplos sensibles debido a la mayor cantidad de muestras de este tipo.



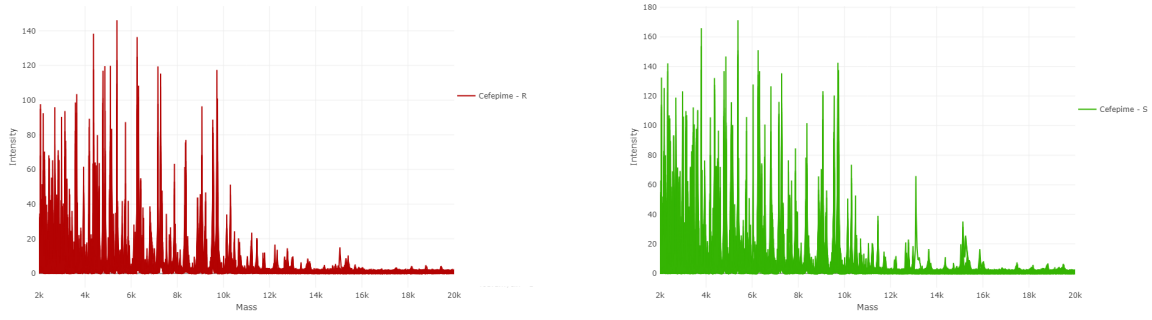
(a) Resistente (b) Sensible

Figura 4.4: Espectros de masas de las muestras resistentes (a) y sensibles (b) al Ciprofloxacin, extraído de [2].



(a) Resistente (b) Sensible

Figura 4.5: Espectros de masas de las muestras resistentes (a) y sensibles (b) al Ceftriaxone, extraído de [2].



(a) Resistente (b) Sensible

Figura 4.6: Espectros de masas de las muestras resistentes (a) y sensibles (b) al Cefepime, extraído de [2]

4.2. Preprocesamiento de los espectros de masas

La etapa de procesado de los espectros de masas se explicó en la Sección 3.1.1, por lo que este punto se centra en determinar las técnicas empleadas en el preprocesado de los conjuntos de datos. Esta etapa se va a llevar a cabo en subconjuntos *train* y *test* por separado, que se detallarán en la Sección 5.2.

En primer lugar, las muestras de *Aspergillus* no pasan por la primera etapa de transformación de los espectros porque los datos han sido tratados con anterioridad, únicamente se lleva a la segunda etapa, búsqueda de picos, para obtener la matriz de picos que será la entrada de los algoritmos de ML. En este caso, en base al estudio [31], se lleva a cabo un alineamiento de los espectros, una detección de picos superiores a un umbral definido en 0.01, se fusionan los picos con distancia inferior a 1 y finalmente, se normalizan por el método TIC.

Por otro lado, el segundo conjunto de datos sobre las muestras de *E. Coli* se trata con las dos etapas de preprocesado, atendiendo al proceso realizado en [19]. En la transformación de los espectros se estabiliza la varianza, se aplica el filtro de suavizado *Savitzky-Golay* con una longitud de ventana de 10 puntos basándose en una regresión polinomial de grado 3. Además, se corrige la línea base y se filtran los valores de masa seleccionando el intervalo comprendido entre 2000 y 20000 *m/z*. Por último, se efectúa la búsqueda de picos mediante un alineamiento de los espectros, una detección de picos superiores al umbral 0.01, se fusionan los picos con distancia inferior a 1 y se normalizan los espectros utilizando el método TIC.

4.3. Detección de valores atípicos

Es de gran importancia evaluar si existen valores atípicos en los conjuntos de datos. La plataforma Clover MS Data Analysis Software utiliza dos técnicas, que se llevan a cabo entre el proceso de transformación de los espectros y el de búsqueda de picos, sobre el conjunto de *train* y se aplica en *test*. Estas técnicas son:

- **Reconstrucción PCA**: consiste en reducir la dimensionalidad de los datos mediante PCA y, a continuación, transformar los datos al espacio inicial. En este proceso de reducción se pierde información, por lo que se calcula el error a partir de la diferencia entre los datos reales y los transformados tras la aplicación de PCA. En este caso, se considera que los valores atípicos serán aquellos cuya tasa de error sea superior a tres veces la desviación típica de la mediana del error.

- **Correlación espectral**: trata de calcular la correlación de cada muestra con todas las demás y obtener la mediana de estos valores. A continuación, se calcula la mediana y la desviación típica de todas las medianas y se representan las muestras mediante un diagrama de cajas. Los valores atípicos tendrán una mediana y desviación típica considerablemente menor que el resto, considerándose en este caso el umbral en 3 veces la desviación típica de la mediana global.

4.3.1. *Aspergillus fumigatus*

En primer lugar, se exponen los resultados de las técnicas utilizadas en la detección de valores atípicos en el conjunto de datos que trata sobre *Aspergillus fumigatus*, y a continuación, se continúa con las muestras de *E.Coli*.

La Figura 4.7 representa mediante diagramas de barras los errores de reconstrucción de cada muestra, la línea horizontal negra corresponde con la mediana global de todas las muestras y la línea discontinua roja sería el valor de 3 desviaciones típicas sobre la mediana global. Se consideran valores atípicos aquellas con un error a una distancia de 3 desviaciones típicas de la mediana del error de reconstrucción global, representándolos de color rojo.

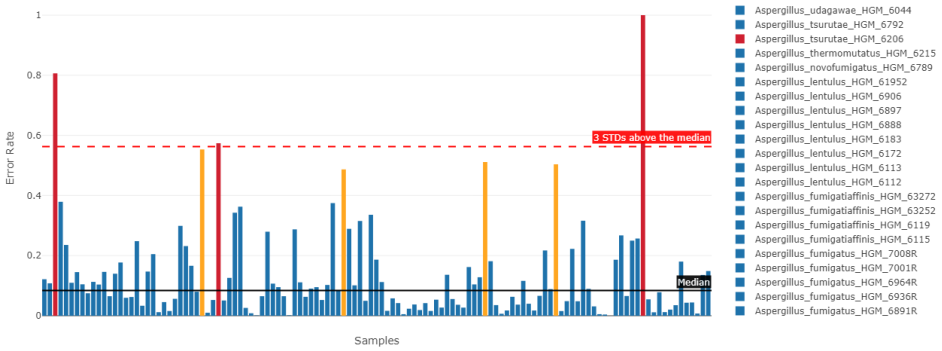


Figura 4.7: Error en la reconstrucción PCA para la detección de valores atípicos en el conjunto de datos de *Aspergillus fumigatus*, extraído de [2]. Las barras verticales representan el error de reconstrucción para cada muestra, siendo de color rojo las consideradas valores atípicos, las amarillas aquellas que están al límite de serlo y de color azul, el resto que no es considerado valor atípico.

Este método define 3 muestras como valores atípicos (de color rojo) por superar la línea discontinua roja, y cuatro muestras al límite de serlo representadas de color amarillo. Por lo

tanto, los valores atípicos según este modelo son ‘Aspergillus_tsurutae_HGM_6206’, ‘Aspergillus_fumigatus_HGM_6567R’ y ‘Aspergillus_fumigatus_HGM_6265S’, que pertenecen a la especie críptica, a la especie *A. fumigatus s.s.* con respuesta resistente a los azoles y a la especie *A. fumigatus s.s.* con respuesta sensible a los azoles respectivamente.

En la Figura 4.8 se exponen las 123 ejemplos del estudio sobre *Aspergillus fumigatus* mediante diagramas de cajas de color azul y rojo. El 50 % intermedio de los datos se muestra en la zona más gruesa de cada diagrama (de color azul o rojo más intenso), la mediana corresponde con el punto medio de la caja y los bigotes que se extienden verticalmente representan el 25 % de los datos superiores y el 25 % de los inferiores. La línea horizontal negra representa el valor de la mediana de todas las medianas y la línea discontinua roja simboliza el valor de 3 desviaciones típicas bajo la mediana global.

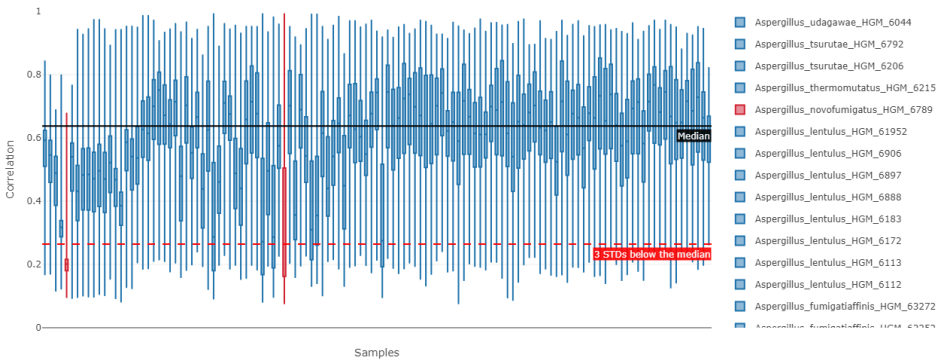


Figura 4.8: Diagrama de cajas de la correlación espectral para la detección de valores atípicos en el conjunto de datos de *Aspergillus fumigatus*, extraído de [2]. De color rojo se representan los diagramas de caja de las muestras consideradas valores atípicos y de color azul las que no.

El método de correlación espectral, por ende, define dos muestras de color rojo como valores atípicos ya que su mediana es bastante inferior a la mediana global, 3 desviaciones típicas bajo la mediana. Estas ejemplos son: ‘Aspergillus_novofumigatus_HGM_6789’ de la especie críptica y ‘Aspergillus_fumigatus_HGM_6444R’ de la especie *A. fumigatus s.s.* resistente a los azoles, estando representadas sus diagramas de cajas en dicho orden en la Figura 4.9.

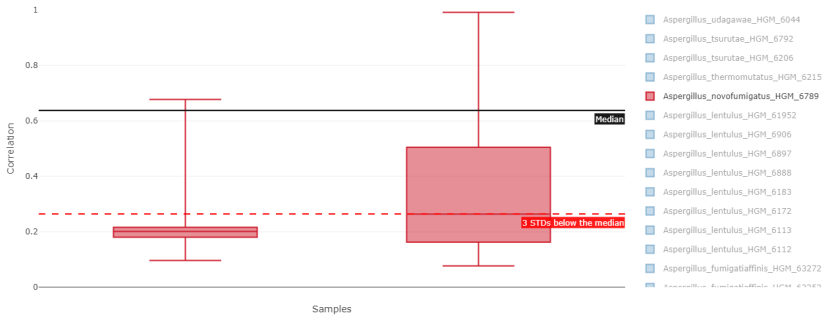


Figura 4.9: Diagrama de cajas de las muestras con valores atípicos detectados mediante Correlación espectral en el conjunto de datos de *Aspergillus*, extraído de [2].

A pesar de que ambos métodos son válidos en la detección de valores atípicos, los resultados en este caso no coinciden, así que, como estas técnicas no son exactas sino que el resultado es aproximado, se considera que los valores atípicos detectados no deberían influir demasiado en el correcto funcionamiento del modelo.

4.3.2. *Escherichia coli*

En este apartado se muestran los resultados de las técnicas explicadas al comienzo de la Sección para las instancias de *E. Coli* que provocan una respuesta resistente o sensible a los antibióticos más relevantes que se tratan en el estudio: Ciprofloxacin, Ceftriaxone y Cefepime.

La Figura 4.10 representa los errores de reconstrucción de cada instancia mediante diagramas de barra. Siete muestras son consideradas como valores atípicos, coloreándose de color rojo, ya que tienen una tasa de error superior a 3 desviaciones típicas de la mediana del error de reconstrucción global. Además, de color amarillo se exponen 2 muestras cuyo error toma un valor cercano al umbral definido. Estas instancias consideradas valores atípicos se exponen en la Tabla 4.2 junto con la respuesta que presentan a los tres antibióticos, desconociéndose la respuesta de las muestras ‘569a8c85’ y ‘1480cb96’ al Ceftriaxone.

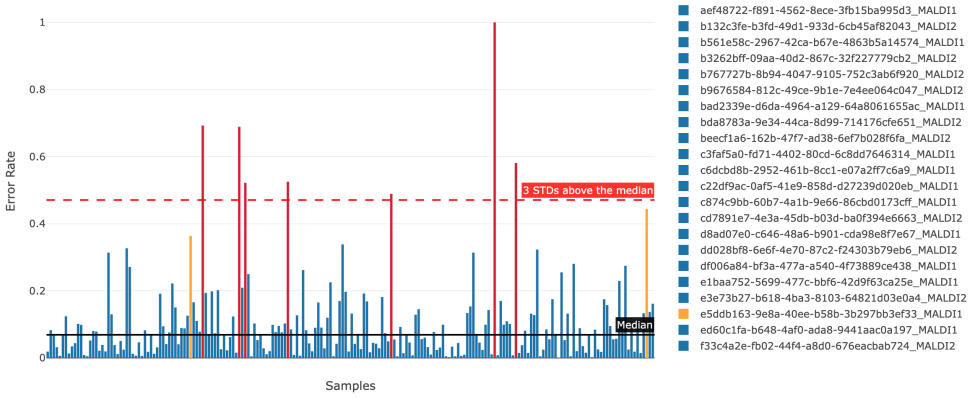


Figura 4.10: Error de la reconstrucción PCA para la detección de valores atípicos en el conjunto de datos de *E. Coli*, extraído de [2]. Las barras verticales representan el error de reconstrucción para cada muestra, siendo de color rojo las consideradas valores atípicos, las amarillas aquellas que están al límite de serlo y de color azul, el resto que no es considerado valor atípico.

Nombre de la muestra	**Respuesta al Ciprofloxacin**	**Respuesta al Ceftriaxone**	**Respuesta al Cefepime**
'1cd3acc52'	Sensible	Sensible	Sensible
'2a6c3609'	Sensible	Sensible	Sensible
'2a88a22b'	Resistente	Sensible	Sensible
'2ef1e5f4'	Sensible	Sensible	Sensible
'3fc3de83'	Sensible	Sensible	Sensible
'569a8c85'	Resistente	-	Resistente
'1480cb96'	Resistente	-	Resistente

Tabla 4.2: Respuesta a los antibióticos de las muestras consideradas valores atípicos por la técnica de reconstrucción PCA.

La Figura 4.11 representa las 208 muestras de la bacteria *E. Coli* mediante diagramas de cajas de color azul y rojo, siendo estas últimas las que poseen una mediana inferior a 3 desviaciones típicas de la mediana global. Por lo tanto, la muestra cuyo nombre comienza por '0dd08acb' es considerada como valor atípico por este método y su diagrama de caja se representa en la Figura 4.12. Esta muestra resulta ser sensible a los antibióticos Ciprofloxacin, Ceftriaxone y Cefepime.

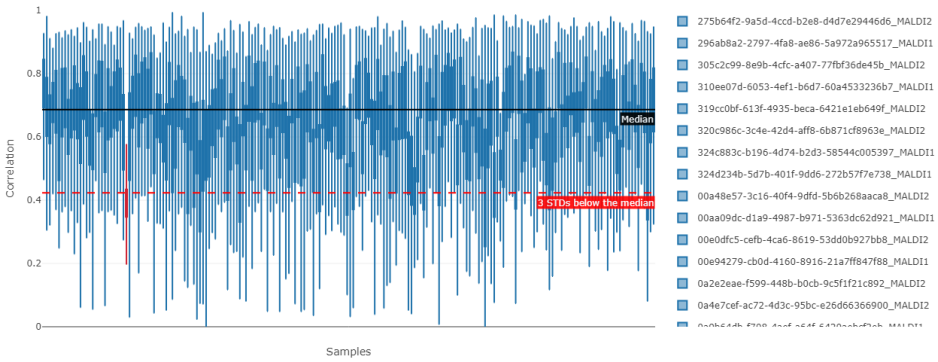


Figura 4.11: Diagrama de cajas de la correlación espectral para la detección de valores atípicos en el conjunto de datos de *E. Coli*, extraído de [2]. De color rojo se representan los diagramas de caja de las muestras consideradas valores atípicos y de color azul las que no.

Figura 4.12: Diagrama de cajas de las muestras con valores atípicos detectados mediante Correlación espectral en el conjunto de datos de *E. Coli*, extraído de [2].

Sin embargo, en este caso, los resultados de ambos métodos tampoco coinciden, considerando ambas técnicas cantidades de valores atípicos muy diferentes (siete muestras frente a una). Así que siguiendo el criterio anterior no se consideran valores atípicos en este conjunto de datos.

Capítulo 5

Experimentos y resultados

En este capítulo se exponen los experimentos realizados a partir de los conjuntos de datos y los métodos descritos a lo largo del TFG con la finalidad principal de identificar la respuesta de las muestras de *Aspergillus fumigatus* a los azoles y de *E. Coli* a los antibióticos Ciprofloxacin, Ceftriaxon y Cefepime, y diseñar modelos de ML que permitan predecir si los microorganismos son resistentes o sensibles a dichos fármacos.

5.1. Software utilizado

Para la realización de este TFG se ha empleado Python (versión 3.8.10) y en concreto se han utilizado las siguientes librerías: `numpy` (versión 1.21.6) y `pandas` (versión 1.1.5), `matplotlib` (versión 3.1.3), `seaborn` (versión 0.11.1) y `plotly` (versión 4.14.3) para el procesamiento de los datos y su visualización, `scikit-learn` (versión 0.23.2) para la aplicación de las herramientas de ML y `torch` (versión 1.13.1), `torchbnnn` (versión 1.2) y `skorch` (versión 0.13.0) para la definición del modelo de BNN.

En el proceso de implementación del código se ha utilizado el entorno de desarrollo integrado Visual Studio Code (versión 1.80.0.) [83], que es un editor de código fuente con herramientas muy útiles como el control integrado de Git, la depuración de código, el resaltado de sintaxis y el autocompletado entre otras. Asimismo, el código se ha distribuido en varios *scripts* de Python según la tarea a realizar por ellos.

5.2. Preparación de los experimentos

Como se explica en la Sección 3.1.2, es fundamental la división del conjunto de datos en los subconjuntos *train* y *test*. En este contexto, se decide aplicar una división del 70 % y el 30 % respectivamente por ser una proporción bastante utilizada. De manera que en el conjunto de *Aspergillus fumigatus* queda distribuido como se muestra en la Tabla 5.1, destinando 123 muestras a la fase de *train* y 52 a la fase de *test*. Asimismo, en la tabla se representa también el tipo de respuesta que las muestras de *Aspergillus fumigatus s.s.* generan a los azoles, desconociéndose en el caso de las especies crípticas. En el conjunto de *E. Coli*, 208 muestras son destinadas para la etapa de entrenamiento y las 88 restantes, para la etapa de evaluación siguiendo la distribución mostrada en la Tabla 5.2. En ambas tablas, se percibe un claro desbalanceo de clases tanto en *train* como en *test*.

Especie	Respuesta a los azoles	Etapa	Nº de muestras	Total
A. fumigatus s.s.	Resistente	*train*	29	43
		test	14	
A. fumigatus s.s.	Sensible	*train*	77	106
		test	29	
Especies crípticas	-	*train*	17	26
		test	9	

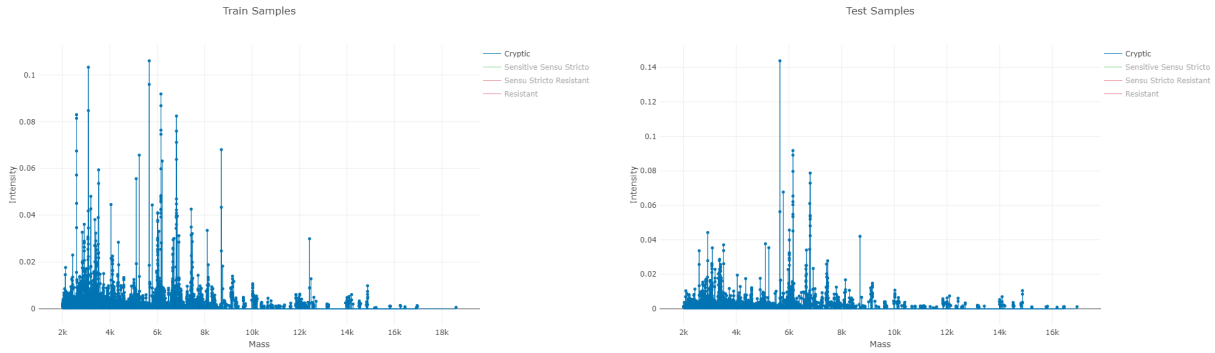
Tabla 5.1: Distribución de las muestras de *A. fumigatus* en los subconjuntos *train* y *test*.

Especie	Respuesta al antibiótico	Etapa	Ciprofloxacin	Ceftriaxone	Cefepime
E. Coli	Resistente	*train*	78	74	31
		test	36	36	19
E. Coli	Sensible	*train*	125	129	138
		test	47	52	55
-	-	Total	286	291	243

Tabla 5.2: Distribución de las muestras de *E. Coli* en los subconjuntos *train* y *test* en función de la respuesta que provoquen ante los antibióticos Ciprofloxacin, Ceftriaxone y Cefepime.

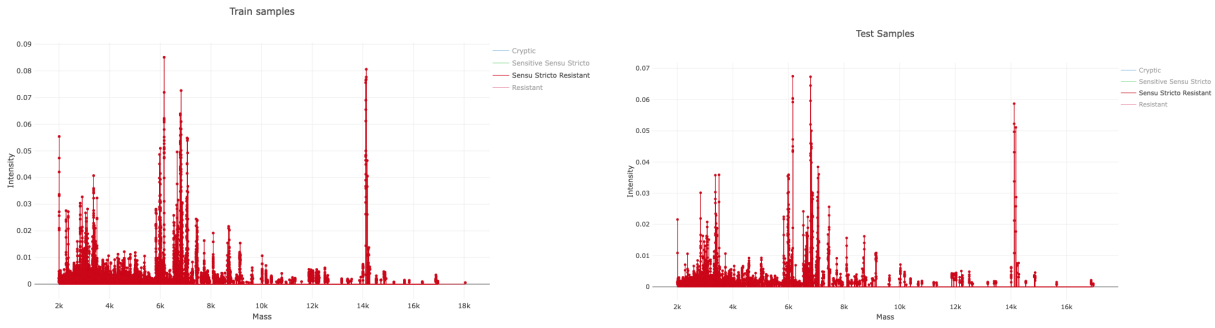
Tras esta división se lleva a cabo el preprocesamiento de los espectros de masas, que queda detallado en la Sección 4.2, obteniéndose una matriz de picos por cada subconjunto *train* y *test* para cada uno de los conjuntos. Las matrices de picos distribuyen la información de la siguiente manera: cada una de las columnas es una muestras y las filas corresponden con la intensidad del pico en los diferentes valores de masa *m/z*. Además se estudia la existencia de los valores atípicos en el subconjunto de *train* para ambos conjuntos (véase en la Sección 4.3).

En la Figura 5.1 se representan gráficamente las matrices de picos correspondientes a los subconjuntos *train* y *test* de las especies crípticas, observándose mayor cantidad de picos y de mayor intensidad en el caso del primer subconjunto. A continuación, la Figura 5.2 expone gráficamente las matrices de picos de los subconjuntos *train* y *test*, propias de la categoría compuesta por las muestras de *A. fumigatus s.s.* con respuesta resistente a los azoles. Se observa que ambos conjuntos presentan una distribución de los picos y de sus intensidades bastante semejantes, mostrándose en el primer subconjunto mayor volumen de picos debido a que está constituido por una cantidad de muestras superior. Finalmente, la Figura 5.3 representa una situación similar para las muestras de *A. fumigatus s.s.* con respuesta sensible a los azoles, en la que ciertos picos tienen mayor intensidad en el subconjunto *train*, pero la distribución de la intensidad de los picos es semejante en ambos.



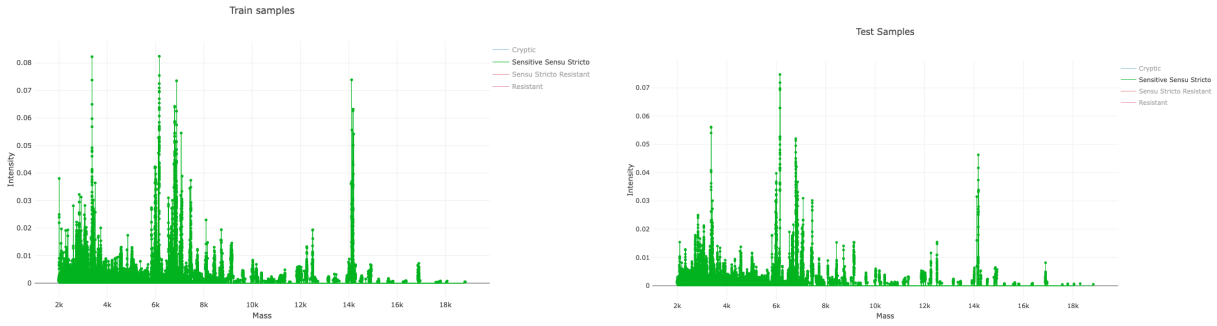
(a) Subconjunto *train* (b) Subconjunto *test*

Figura 5.1: Representación de las matrices de picos extraídas de los subconjuntos *train* (a) y *test* (b) del conjunto de muestras de las especies crípticas, extraído de [2].



(a) Subconjunto *train* (b) Subconjunto *test*

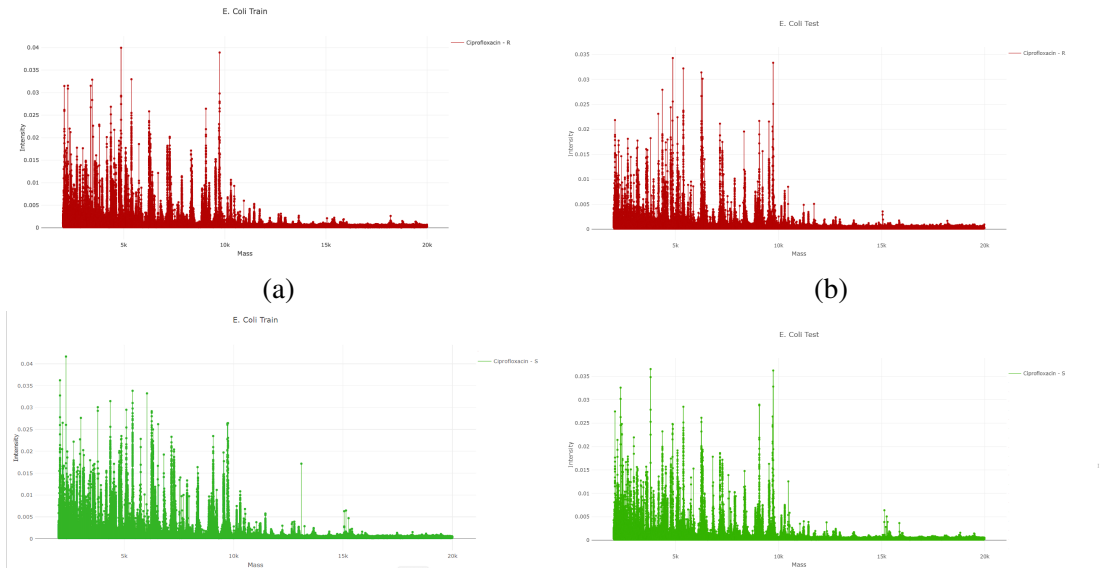
Figura 5.2: Representación de las matrices de picos extraídas de los subconjuntos *train* (a) y *test* (b) del conjunto de muestras de la especie *A. fumigatus s.s.* con respuesta resistente a los azoles, extraído de [2].



(a) Subconjunto *train* (b) Subconjunto *test*

Figura 5.3: Representación de las matrices de picos extraídas de los subconjuntos *train* (a) y *test* (b) del conjunto de muestras de la especia *A. fumigatus s.s.* con respuesta sensible a los azoles, extraído de [2].

Asimismo, la Figura 5.4 manifiesta gráficamente las matrices de picos de las muestras de *E. Coli* que generan una respuesta resistente o sensible al antibiótico Ciprofloxacin, separadas en los subconjuntos de *train* y *test*. En ella, se observa que en este contexto, la disposición de los picos también es muy semejante entre los subconjuntos *train* y *test* para ambos tipos de respuesta.



(a) (b)

(c) (d)

Figura 5.4: Representación de las matrices de picos del conjunto de muestras de la especie *E. Coli* con respuesta al Ciprofloxacin, extraído de [2]. (a) matriz de picos del subconjunto *train* con resistencia al Ciprofloxacin; (b) matriz de picos del subconjunto *test* con resistencia al Ciprofloxacin; (c) matriz de picos del subconjunto *train* con sensibilidad al Ciprofloxacin; (d) matriz de picos del subconjunto *test* con Sensibilidad al Ciprofloxacin.

La Figura 5.5 representa gráficamente las matrices de picos de las muestras de *E. Coli* que generan una respuesta resistente o sensible al antibiótico Ceftriaxone, separadas en los subconjuntos de *train* y *test*. En ella, se contempla que la disposición de los picos también es muy similar entre los subconjuntos *train* y *test* para ambos tipos de respuesta, aunque en las muestras resistentes, se observan algunos picos de mayor intensidad en el subconjunto *test*. Además se observa un mayor volumen de picos en la representación asociada a las matrices de picos del subconjunto *train*.

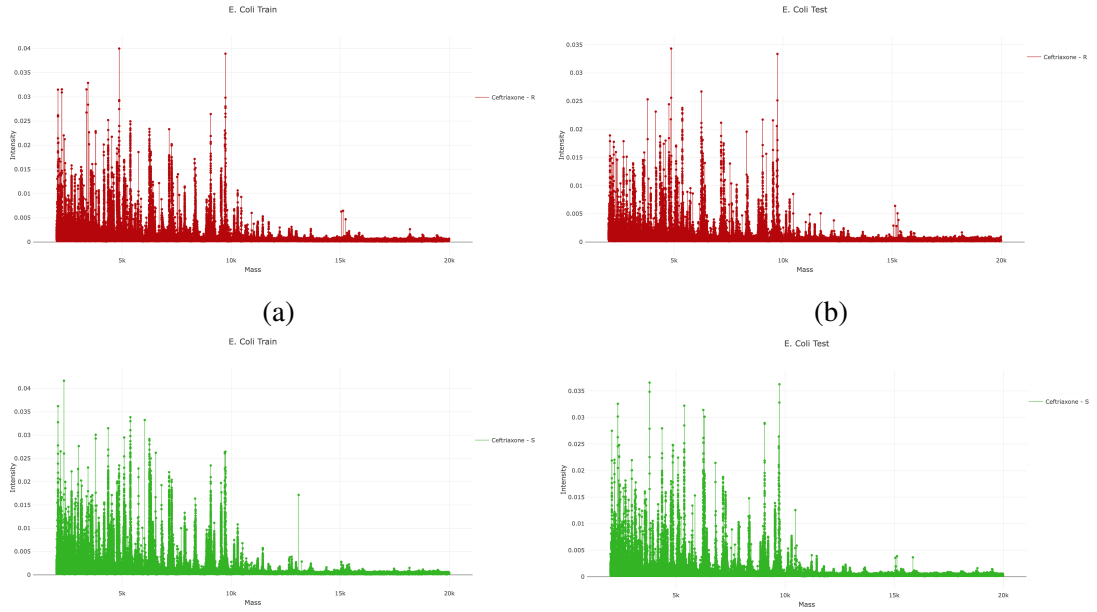


Figura 5.5: Representación de las matrices de picos del conjunto de muestras de la especie *E. Coli* con respuesta al Ceftriaxone, extraído de [2]. (a) matriz de picos del subconjunto *train* con resistencia al Ceftriaxone; (b) matriz de picos del subconjunto *test* con resistencia al Ceftriaxone; (c) matriz de picos del subconjunto *train* con sensibilidad al Ceftriaxone; (d) matriz de picos del subconjunto *test* con Sensibilidad al Ceftriaxone.

Finalmente, la Figura 5.6, que representa las matrices de picos correspondientes para el antibiótico Cefepime, expone mayor diferencia en la disposición de los picos que las anteriores (Figuras 5.4 y 5.5) entre los subconjuntos *train* y *test*.

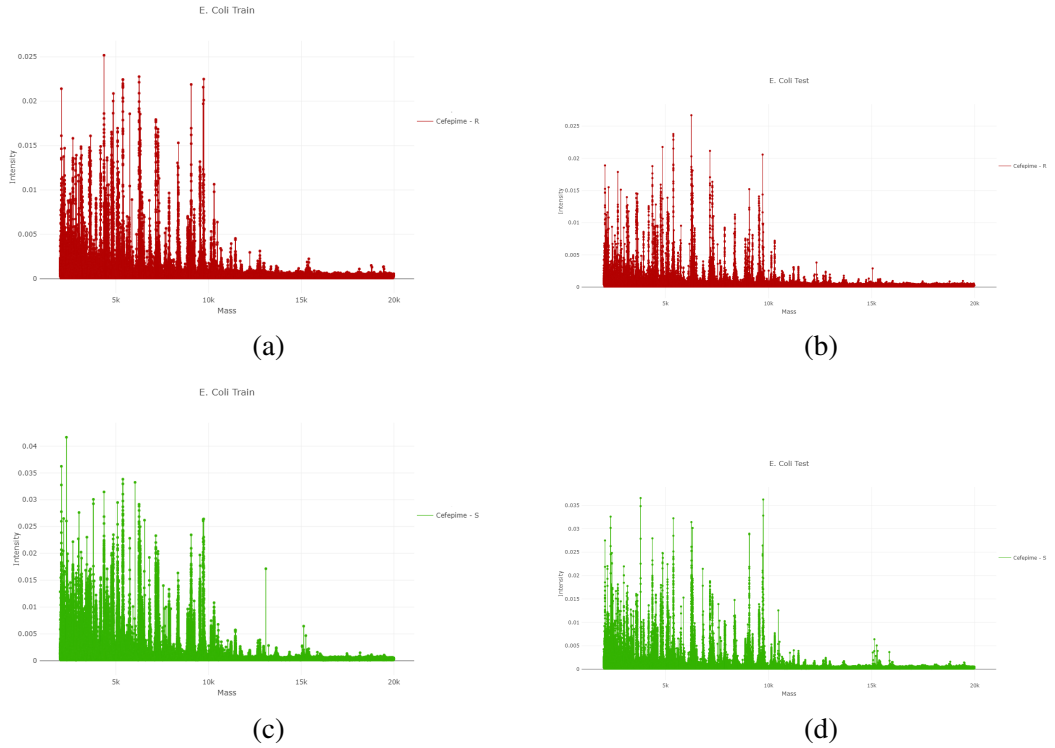


Figura 5.6: Representación de las matrices de picos del conjunto de muestras de la especie *E. Coli* con respuesta al Cefepime, extraído de [2]. (a) matriz de picos del subconjunto *train* con resistencia al Cefepime; (b) matriz de picos del subconjunto *test* con resistencia al Cefepime; (c) matriz de picos del subconjunto *train* con sensibilidad al Cefepime; (d) matriz de picos del subconjunto *test* con Sensibilidad al Cefepime.

Adicionalmente, se observa que, a pesar de haber reducido la dimensionalidad de ambos conjuntos de datos con la etapa de búsqueda de picos, cuyo resultado es la matriz de picos, siguen existiendo un número elevado de variables (picos) que definen las muestras. En el conjunto de *Aspergillus fumigatus* se ha reducido a 1588 valores de masas espectrales *m/z* y en el caso de *E. Coli* a 5658 valores de masas espectrales, partiendo ambos de 18000 valores de masas espectrales debido al rango seleccionado anteriormente en el preprocesamiento de los espectros de masas, [2000, 20000] *m/z*. Para minimizar este volumen de características se aplica el método PCA, explicado en la Sección 3.1.4, que comprime los conjuntos reduciendo la cantidad de variables usadas en la definición del modelo BNN. El número de componentes en PCA se obtiene tras calcular la varianza acumulativa de cada uno de los componentes y se seleccionan aquellos con un valor de la varianza acumulada entre el 80 y el 85 %.

Para poder utilizar las matrices de picos como entrada de un modelo de ML, los ejemplos deben situarse en las filas y cada columna debe ser una característica de las instancias. El resul-

tado se guarda en una matriz para cada subconjunto, con el objetivo de aplicar el algoritmo de PCA sobre el subconjunto de *train*, calculando la varianza explicada de cada uno de los componentes y eligiendo aquellos componentes con un valor alto de varianza acumulada (entre el 80 y el 85 %) ya que serán aquellos más significativos. Éstos serán los componentes que tomaremos como características para la implementación de los modelos BNN, RF y LightGBM. A continuación, el conjunto de *test* se transforma en un conjunto de variables reducidas conforme al conjunto de *train*. De esta manera, se obtienen en ambos conjuntos un vector de características reducido con las mismas dimensiones (véase en la Tabla 5.3).

En la Tabla 5.3 quedan reflejadas las dimensiones de la matriz distinciones posibles entre las muestras de los conjuntos de datos de *A. fumigatus* y *E. Coli*, detalladas en la Sección 5.4. El número de variables, reflejado en la última columna, es el resultado obtenido tras la aplicación de la técnica PCA, habiéndose reducido el número de variables propias de la matriz de picos en ambos conjuntos.

Conjunto de datos	**Distinción entre clases**	**Antimicrobiano**	**Nº de muestras**	**Nº de variables**
	s.s. vs crípticas	Azoles	175	20
A. Fumigatus	s.s. Resistente vs Sensible	Azoles	149	17
	Resistente vs Sensible	Ciprofloxacin	286	73
	Resistente vs Sensible	Ceftriaxone	291	72
E. Coli	Resistente vs Sensible	Cefepime	243	66

Tabla 5.3: Distinción entre las distintas clases que forman los conjuntos de datos de *A. fumigatus* y *E. Coli* junto con sus dimensiones y los antimicrobianos a los que presentan resistencia.

En ambos conjuntos de datos, las clases están claramente desbalanceadas, como mostraban las Tablas 5.1 y 5.2. Sin embargo, se decide no aplicar ningún método para tratarlo debido a que lo normal es que la mayoría de las muestras, tanto de *A. fumigatus* [34] como de *E. Coli* [84], sean sensibles a los antimicrobianos correspondientes. Aún así este desbalanceo se tiene en cuenta, la fase de entrenamiento del modelo se lleva a cabo utilizando la técnica *Statrified K-Fold CV*, que selecciona las muestras para cada *fold* siguiendo la proporción en la que encuentran en el conjunto de datos, y la métrica utilizada en el proceso de optimización de hiperparámetros es *balanced accuracy*.

Finalmente, para que los conjuntos de datos estén completamente preparados para la definición del modelo de BNN, las matrices deben transformarse en tensores, que son una estructura de datos que representa matrices multidimensionales en PyTorch [85], la biblioteca principal empleada en la implementación del algoritmo.

5.3. Implementación de la BNN

Las librerías ampliamente utilizadas en lenguaje Python para el diseño de ANN son PyTorch [85], TensorFlow [86] y Keras [87]. En el presente TFG, con el objetivo de desarrollar una BNN que desempeña una tarea de clasificación, se ha utilizado PyTorch con la ayuda de la librería `torchbnn` que extiende las capacidades de la primera añadiendo capas bayesianas lineales y convolucionales. En este caso, se trabaja con capas bayesianas lineales, pudiendo ser interesante el futuro estudio de con capas bayesianas convolucionales.

La arquitectura de la red contiene una primera capa bayesiana lineal (*Bayesina Linear Layer* en inglés) compuesta de la cantidad de neuronas resultante tras la optimización del hiperparámetro *hidden neurons* mediante una *Stratified 5-Fold CV* (detallado en la Sección 5.4). Su entrada se define por las n características obtenidas de la técnica PCA y la salida de esta primera capa se obtiene a partir de la función de activación ReLU, que anula los valores negativos introducidos y mantiene los positivos. A continuación la salida de esta capa supone la entrada de la segunda capa bayesiana lineal, capa de salida. Está formada por tantas neuronas como clases a identificar, es decir, dos, y una función de activación que se optimiza. La Figura 5.7 manifiesta un ejemplo de la arquitectura en la que la función de activación de la primera capa bayesiana lineal es ReLU, que confirma que las características entrantes tienen un valor 0 o superior, y en la segunda, Softmax, que transforma la salida en una distribución de probabilidades que suman el valor 1.

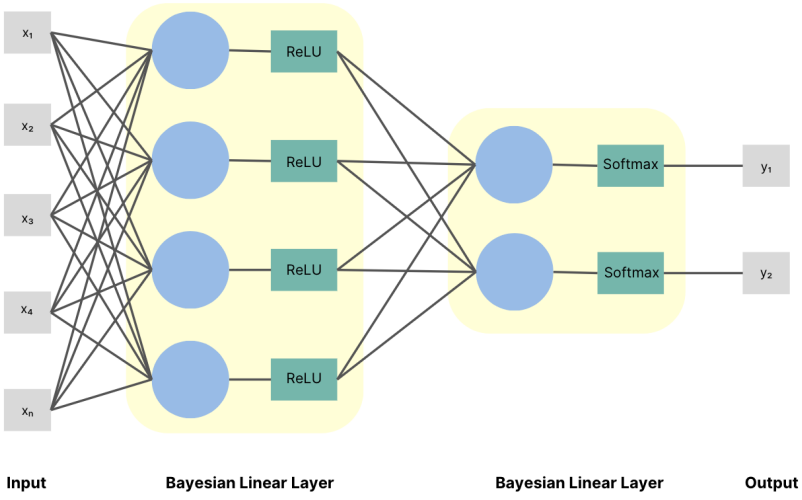


Figura 5.7: Ejemplo de arquitectura de la BNN diseñada en la que tras el proceso de optimización de hiperparámetros, se seleccionan las funciones de activación ReLU y Softmax (elaboración propia).

Durante la etapa de entrenamiento de la BNN, se lleva a cabo el proceso de retropropagación una cantidad máxima de veces marcada por el hiperparámetro *max epochs*. En cada una de las iteraciones, la red procesa la cantidad de muestras de entrada definida por el hiperparámetro *batch size* y calcula el coste. Éste informa cuánto se aleja la etiqueta predicha de la etiqueta real a partir de la definición de una función de coste que relaciona la pérdida de entropía cruzada con la pérdida binaria de Kullback-Leiber. Además, se utiliza el optimizador Adam con una tasa de aprendizaje de 0.01 para ajustar los pesos de la red minimizando el coste.

En este modelo, se decide optimizar los hiperparámetros *batch size, max epochs, activation function* y *hidden neurons*, explicados en la Sección 3.2.2, mediante un proceso de CV del tipo *Stratified 5-Fold*, que utiliza como métrica *balanced accuracy* debido al desbalanceo de clases. Para el hiperparámetro *batch size* se explora un rango comprendido entre 10 y 50 en pasos de 10 unidades. El hiperparámetro *max epochs* puede tomar como valor 10, 50 o 100 y *hidden neurons* explora los valores múltiplos de 10 dentro del intervalo comprendido entre 40 y 120. Finalmente, la *activation function* puede ser Sigmoid o Softmax puesto que al ser una clasificación binaria, la salida debe estar en un rango [0,1] y con estos tipos de función de activación se consigue. Tras la exploración con todas las posibles combinaciones, el modelo se entrena con aquella que presente el mejor valor para la métrica *balanced accuracy*, detallándose en la Sección 5.4.

Para llevar a cabo el proceso de optimización de hiperparámetros se utiliza de la librería `skorch` [88], ya que permite al modelo creado con PyTorch ser compatible con la biblioteca `scikit-learn` [89], que ofrece métodos utilizados comúnmente que facilitan la exploración de los diferentes resultados ante una matriz de parámetros de entrada. Por lo tanto, se define un modelo clasificador de BNN con `skorch` en el que se fijan los argumentos *Criterion* y *Optimizador* con su correspondiente tasa de aprendizaje y se dejan libres los hiperparámetros que se pretende optimizar.

El algoritmo creado se entrena con el subconjunto de muestras de *train* junto con los mejores hiperparámetros y a continuación, se evalúa clasificando las muestras del subconjunto de *test*, obteniendo los resultados que se exponen en la siguiente Sección.

5.4. Resultados

En el presente TFG se implementa una BNN cuyos resultados se comparan con los obtenidos a partir de los modelos RF y LightGBM, definidos en las Secciones 3.2.4 y 3.2.5 respectivamente.

Los clasificadores RF y LightGBM toman como conjuntos de entrada, los mismos que se han definido al comienzo de esta sección, con la misma división en los subconjuntos *train* y *test* y el mismo preprocesamiento de los espectros de masas. Además, para cada uno de los clasificadores se optimizan los hiperparámetros de los modelos detallados en las Secciones 3.2.4 y 3.2.5 respectivamente, utilizándose el resto de parámetros definidos por defecto en la documentación de los modelos RF [90] y LightGBM [91] con alguna excepción. En el caso de RF, los parámetros *warm start* y *oob score* son definidos como *True* por Clover, modificando los estimadores del bosque creado con anterioridad y permitiendo calcular la métrica OOB (del inglés *out-of-bag*) en las muestras no utilizadas para el entrenamiento. Por otro lado, al parámetro *min samples per leaf* se le permite tomar valores entre 2 y 20 para evitar errores cuando la cantidad muestras del conjunto de datos sea menor que 20 (siendo 20 el valor por defecto para este parámetro en la definición de LightGBM).

5.4.1. Discriminación de especie *A. fumigatus s.s.* y especies crípticas

En esta subsección, se trata de exponer los resultados obtenidos en el diseño de los modelos discriminando las muestras del conjunto *A. fumigatus* en la especie *A. fumigatus s.s.* y las especies crípticas.

Entrenamiento del modelo BNN con *Stratified* 5-*Fold CV*

En la fase de entrenamiento de la BNN, se toman las muestras del subconjunto de *train* (123 instancias) de las que 106 pertenecen a la especie *A. fumigatus s.s.* y las 17 restantes a las especies crípticas. Tras la optimización de los hiperparámetros, se obtienen que aquellos que proporcionan la mejor predicción son los expuestos en la Tabla 5.4. Por lo tanto, el algoritmo de retropropagación se ha ejecuta un máximo de 50 veces (*max epochs*) utilizando 10 muestras por iteración (*batch size*). En cuanto a la arquitectura de la red, contamos con 20 neuronas de entrada, 70 neuronas ocultas (*hidden neurons*) y 2 neuronas de salida. La cantidad de neuronas de entrada viene dada por el número de características que definen las muestras y la de neuronas de salida, por la cantidad de clases. Además, la función de activación de la primera capa bayesiana lineal es ReLU y la seleccionada para la segunda capa bayesiana es Sigmoid.

Hiperparámetro	**Valor optimizado**
batch size	10
max epochs	50
hidden neurons	70
activation function layer	Sigmoid

Tabla 5.4: Valor de los hiperparámetros optimizados de la BNN para la discriminación de especies *A. fumigatus s.s.* y especies crípticas.

Evaluación del modelo BNN

Con el objetivo de evaluar el modelo de BNN, se separaron 52 muestras del total (subconjunto de *test*), de las cuales 43 pertenecen a la especie *A. fumigatus s.s.* y las 9 restantes a las especies crípticas. Se ha obtenido como resultado una *accuracy* del 96.15 %, una *precision* del 95.56 % , una *recall* del 100 %, una *specificity* del 77.78 % y una *balanced accuracy* del 88.89 %. Estos valores se obtienen a partir de la matriz de confusión representada en la Tabla 5.5. Además, se obtiene un valor de AUC igual a 0.889.

		Valor Clasificación	
		Especies crípticas	*A. fumigatus s.s.*
Valor Real	Especies crípticas	7	2
	A. fumigatus s.s.	0	43

Tabla 5.5: Matriz de confusión para discriminación de especies *A. fumigatus s.s.* y especies crípticas por la BNN.

Finalmente, en la Figura 5.8 se expone las 52 muestras a evaluar diferenciadas en dos colores según la clase y exponiendo una comparativa entre la clasificación real y la predicha por la BNN. La especie *A. fumigatus s.s.* es de color amarillo y las especies crípticas son de color morado. Se observa que la BNN ha predicho de forma errónea dos muestras de las especies crípticas como *A. fumigatus s.s.*

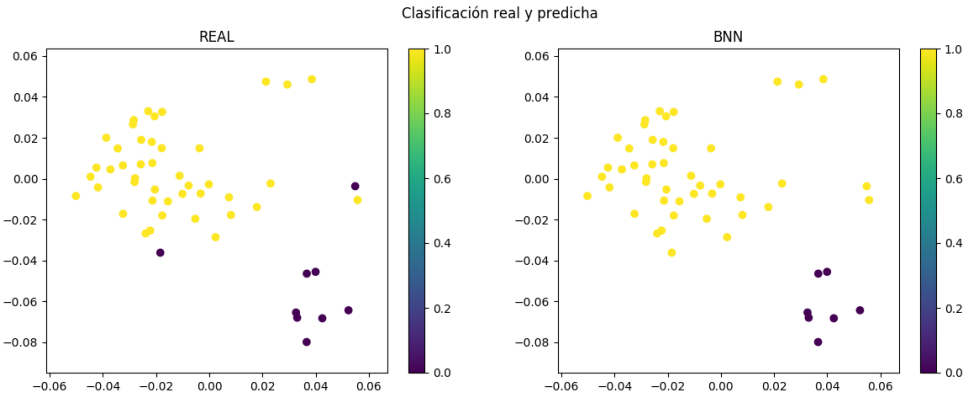


Figura 5.8: Representación gráfica en 2 dimensiones de la discriminación de especies *A. fumigatus s.s.* (de color amarillo) y especies crípticas (de color morado) por la BNN.

Comparación con los algoritmos RF y LightGBM

Los modelos RF y LightGBM se ejecutan con la combinación de hiperparámetros que obtienen el valor más alto de la métrica *balanced accuracy*, mostrados en la Tabla 5.6. El modelo de RF ha sido diseñado con 50 árboles de decisión (*estimators*), con una longitud máxima entre el nodo raíz y una hoja de 10 (*Max Depth*), con al menos 1 muestra por hoja (*Min Samples per leaf*), con un número máximo de 39 características a tener en cuenta en cada árbol (*Max Features*) y un número mínimo de 2 instancias para dividir el nodo (*Min Split size*). En el caso de LightGBM, este algoritmo ha sido implementado con 200 árboles (*estimators*), una tasa de 0.01 para minimizar el sobreajuste (*learning rate*), una profundidad de 8 nodos hojas (*number of leaves*) y un número mínimo de 12 hojas por árbol (*Min samples per leaf*).

Hiperparámetro	**Valor optimizado**
Estimators	50
Max depth	10
Min samples per leaf	1
Max features	39
Min samples split	2

Hiperparámetro	**Valor optimizado**
Estimators	200
Learning rate	0.01
Number of leaves	8
Min samples per leaf	12

Tabla 5.6: Hiperparámetros optimizados en el entrenamiento de los algoritmos RF (izquierda) y LightGBM (derecha) para la discriminación de especies *A. fumigatus s.s.* y especies crípticas.

La Tabla 5.7 muestra las figuras de mérito obtenidas con los tres clasificadores diseñados en este TFG. En ella se observa que el modelo con mejores prestaciones es la BNN. Además, es importante resaltar que se registra un valor de la figura de mérito *specificity* bastante superior

en el caso de la BNN con respecto a RF y LightGBM, lo que indica que la red es más confiable que el resto de modelos en la clasificación de la clase minoritaria (especies crípticas), fallando en la categorización de 2 de las 9 totales.

Algoritmo	***Accuracy***	***Precision***	***Recall***	***Specificity***	***Balanced Accuracy***	**AUC**
RF	86.54 %	87.50 %	97.67 %	33.33 %	65.50 %	0.845
LightGBM	84.62 %	87.23 %	95.35 %	33.33 %	64.34 %	0.806
BNN	**96.15 %**	**95.56 %**	**100 %**	**77.78 %**	**88.89 %**	**0.889**

Tabla 5.7: Comparación de las métricas obtenidas en los distintos algoritmos para la discriminación de especies *A. fumigatus s.s.* y especies crípticas. Se muestran en negrita los mejores resultados.

5.4.2. Discriminación de especies *A. fumigatus s.s.* resistentes y sensibles

A continuación, se trata de clasificar con BNN, RF y LightGBM las muestras de la especie *A. fumigatus s.s.* que presentan una respuesta de resistencia o sensibilidad a los azoles.

Entrenamiento del modelo BNN con *Stratified* 5-*Fold CV*

En la fase de entrenamiento del modelo, de las 123 muestras totales destinadas a ésta, se toma las 106 pertenecientes a la especie *Aspegillus fumigatus* que se dividen en 29 instancias del tipo resistente y 77 del tipo sensible.

Los hiperparámetros que logran la mejor predicción son los expuestos en la Tabla 5.8. De manera que el modelo de BNN implementado ha ejecutado el algoritmo *backpropagation* un máximo de 100 veces (*max epochs*) separando las muestras en 10 por iteración (*batch size*). Además se han utilizado 40 neuronas ocultas (*hidden neurons*) y la función de activación en la primera capa es ReLU, y en la segunda (*activation function*) se determina la función de activación Sigmoid.

Hiperparámetro	**Valor optimizado**
batch size	10
max epochs	100
hidden neurons	40
activation function	Sigmoid

Tabla 5.8: Valor de los hiperparámetros optimizados en la BNN para la discriminación de especies *A. fumigatus s.s.* resistentes y sensibles.

Evaluación del modelo BNN

En la fase de evaluación se utilizan 43 muestras del subconjunto de *test*. De esta división, 14 presentan una respuesta resistente a los azoles y las 29 restante presentan una respuesta sensible a dichos antifúngicos. Como resultado de esta fase, se obtiene la matriz de confusión expuesta en la Tabla 5.9 de la que se obtienen unos valores del 100 % para las figuras de mérito *accuracy*, *precision*, *recall*, *specificity* y *balanced accuracy* y una AUC de 1. Estos resultados se deben a que la BNN clasifica correctamente todas las muestras utilizadas en esta evaluación del modelo creado.

		Valor Clasificación	
		Sensible	Resistente
Valor Real	Sensible	29	0
	Resistente	0	14

Tabla 5.9: Matriz de confusión para discriminación de especies *A. fumigatus s.s.* resistentes y sensibles por la BNN.

Por último, la Figura 5.9 expone la comparativa entre la clasificación real y la predicha por la BNN de los ejemplos del subconjunto de *test*. Las instancias de *A. fumigatus s.s.* resistentes se colorean de amarillo y las sensibles de muestran color morado. Por lo tanto, se observa cómo todas muestras han sido predichas correctamente por la BNN.

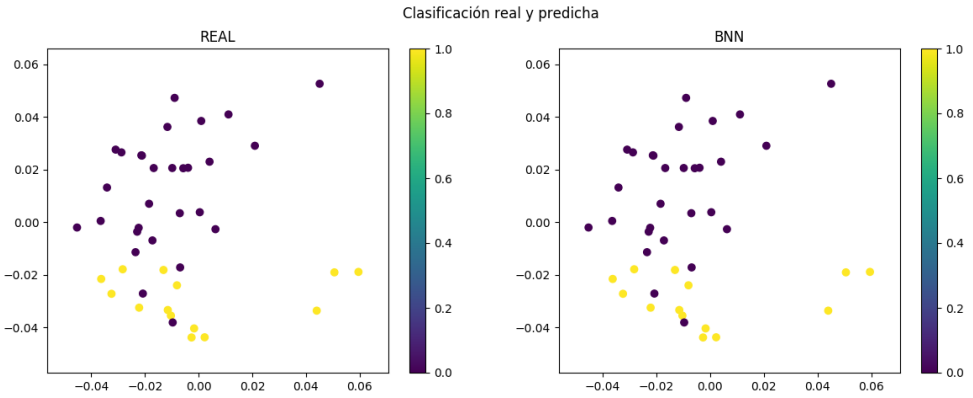


Figura 5.9: Representación gráfica en 2 dimensiones de la discriminación de especies *A. fumigatus s.s.* resistentes (de color amarillo) y sensibles (de color morado) por la BNN.

Comparación con los algoritmos RF y LightGBM

Tras la optimización de los hiperparámetros de RF y LightGBM, se muestran en la Tabla 5.10 las combinaciones que obtienen un valor de *balanced accuracy* mayor. Por un lado, se obtiene un diseño del modelo RF compuesto por 200 árboles de decisión (*estimators*), con una longitud máxima entre el nodo raíz y una hoja de 10 (*Max Depth*), con al menos 4 muestras por hoja (*Min Samples per leaf*), con un número máximo de 39 características a considerar en cada árbol (*Max Features*) y un número mínimo de 2 instancias para dividir el (*Min Split size*). Por otro lado, LightGBM ha sido implementado con 200 árboles (*estimators*), una tasa de 0.1 para minimizar el sobreajuste (*learning rate*), una profundidad de 8 nodos hojas (*number of leaves*) y un número mínimo de 10 hojas por árbol (*Min samples per leaf*).

Hiperparámetro	**Valor optimizado**
Estimators	200
Max Depth	10
Min Samples per leaf	4
Max Features	39
Min Split size	2

Hiperparámetro	**Valor optimizado**
Estimators	200
Learning rate	0.1
Number of leaves	8
Min samples per leaf	10

Tabla 5.10: Hiperparámetros optimizados en el entrenamiento de los algoritmos RF (izquierda) y LightGBM (derecha) para la discriminación de especies *A. fumigatus s.s.* resistentes y sensibles.

En la Tabla 5.11 se refleja las prestaciones de los modelos RF, LightGBM y BNN, viéndose que el modelo con mejores valores para las figuras de mérito es la BNN puesto que es el único que predice las 43 correctamente.

Algoritmo	***Accuracy***	***Precision***	***Recall***	***Specificity***	***Balanced Accuracy***	**AUC**
RF	72.09 %	75 %	21.43 %	96.55 %	58.99 %	0.874
LightGBM	83.72 %	73.33 %	78.57 %	86.21 %	82.39 %	0.909
BNN	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**1**

Tabla 5.11: Comparación de las métricas obtenidas en los distintos algoritmos para la discriminación de especies *A. fumigatus s.s.* resistentes y sensibles. Se muestran en negrita los mejores resultados.

5.4.3. Discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ciprofloxacin

En este apartado se pretende categorizar las muestras de la bacteria *E. Coli* en función de su respuesta ante el antibiótico Ciprofloxacin (resistente o sensible).

Entrenamiento del modelo BNN con *Stratified* 5-*Fold CV*

En este caso, el conjunto de datos es el de las bacterias del tipo *E. Coli*. Para entrenar los modelos, de las 208 muestras que pertenecen al subconjunto *test*, se toman 203 puesto que son aquellas que generan respuesta al antibiótico Ciprofloxacin. 78 resultan ser resistente a este medicamento y 125 sensibles.

Para llevar a cabo la definición del modelo, se optimizan los hiperparámetros que refleja la Tabla 5.12. Por lo tanto, la BNN implementada ha ejecutado el algoritmo *backpropagation* un máximo de 100 veces (*max epochs*) separando las muestras en 10 por iteración (*batch size*). Además se han utilizado 110 neuronas ocultas (*hidden neurons*) y la función de activación (*activation function*) de la última capa ha sido Sigmoid.

Hiperparámetro	Valor optimizado
batch size	10
max epochs	100
hidden neurons	110
activation function	Sigmoid

Tabla 5.12: Valor de los hiperparámetros optimizados en la BNN para la discriminación de bacterias *E. Coli* resistentes y sensibles al Ciprofloxacin.

Evaluación del modelo BNN

De las 88 muestras que forman el subconjunto de *test*, 36 son resistentes al Ciprofloxacin, 47 son sensibles y el resto no generan ninguna respuesta ante el antibiótico. Tomando estas instancias, el modelo de BNN trata de clasificarlas y se obtiene la matriz de confusión representada en la Tabla 5.13. A partir de ella se calculan las figuras de mérito y se obtiene una *accuracy* del 68.67 %, una *precision* del 69.81 % , una *recall* del 78.72 % , una *specificity* del 55.56 % y una *balanced accuracy* del 67.13 %. Además, se obtiene un valor de AUC de 0.671.

		Valor Clasificación	
		Sensible	Resistente
Valor Real	Sensible	37	10
	Resistente	16	20

Tabla 5.13: Matriz de confusión para discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ciprofloxacin por la BNN.

Asimismo, la Figura 5.10 representa gráficamente la comparación entre la clasificación real y la predicha por el modelo, mostrando de color amarillo las muestras de la clase sensible y de color morado las de la clase resistente.

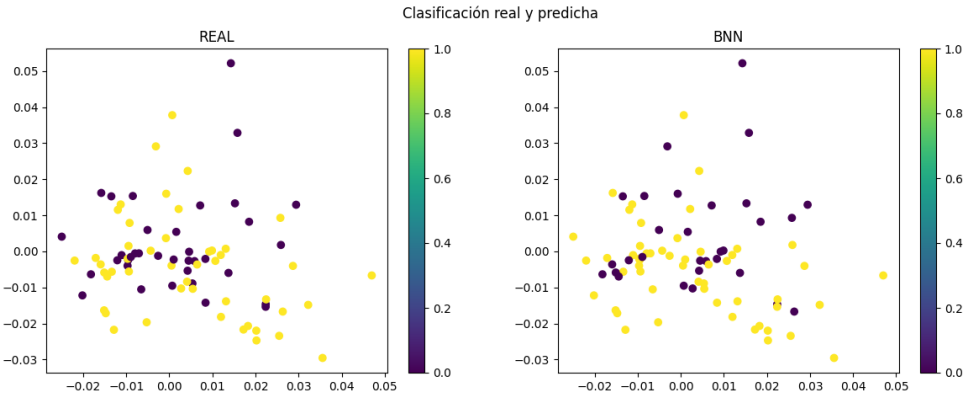


Figura 5.10: Representación gráfica en 2 dimensiones de la discriminación de bacterias *E. Coli* resistentes (de color morado) y sensibles (de color amarillo) al antibiótico Ciprofloxacin.

Comparación con los algoritmos RF y LightGBM

La Tabla 5.14 representa los hiperparámetros optimizados en los modelos RF y LightGBM, de manera que el RF ha sido creado con 100 árboles de decisión, teniendo una profundidad de 30, con al menos 2 muestras por hoja, con un máximo de 75 características por árbol y un número mínimo de 5 observaciones para dividir el nodo. Asimismo, el modelo LightGBM se define con 100 árboles, una tasa de aprendizaje de 0.01, una profundidad de 8 nodos hojas y al menos 20 hojas por árbol.

Hiperparámetro	Valor optimizado
Estimators	100
Max Depth	30
Min Samples per leaf	2
Max Features	75
Min Split size	5

Hiperparámetro	Valor optimizado
Estimators	100
Learning rate	0.01
Number of leaves	8
Min samples per leaf	20

Tabla 5.14: Hiperparámetros optimizados en el entrenamiento de los algoritmos RF (izquierda) y LightGBM (derecha) respectivamente para la discriminación de especies *E. Coli* resistentes y sensibles al Ciprofloxacin.

Acorde con la Tabla 5.15, en la que se recogen las prestaciones de los modelos RF, LightBM y BNN, el que obtiene mejores resultados globales es LightGBM puesto que los valores de las métricas *balanced accuracy* y AUC son superiores. Sin embargo, la BNN resulta ser el modelo que mejor clasifica a la clase negativa (resistente) correctamente (resistente) debido a obtener el valor más alto en la métrica *specificity*.

Algoritmo	*Accuracy*	*Precision*	*Recall*	*Specificity*	*Balanced Accuracy*	AUC
RF	59.04 %	59.15 %	89.36 %	19.44 %	54.4 %	0.648
LightGBM	**71.08 %**	66.67 %	**97.87 %**	36.11 %	**66.99 %**	**0.694**
BNN	68.67 %	**69.81** %	78.72 %	**55.56 %**	55.56 %	0.671

Tabla 5.15: Comparación de las métricas obtenidas en los distintos algoritmos para la discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ciprofloxacin. Se muestran en negrita los mejores resultados.

5.4.4. Discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ceftriaxone

Se lleva a cabo la clasificación de las muestras de *E. Coli* que presentan una respuesta sensible o resistente al antibiótico Ceftriaxone.

Entrenamiento del modelo BNN con *Stratified* 5-*Fold CV*

En la etapa de *train* se toman 203 muestras de las que 74 tienen una respuesta resistente al antibiótico Ceftriaxone, y las 129 restantes son sensibles al medicamento. Los hiperparámetros optimizados del modelo de BNN se reflejan en la Tabla 5.16, mostrando que el algoritmo de retropropagación se ejecuta un máximo de 100 veces, de manera que en cada iteración se utilizan

20 muestras. Además, la red cuenta con 100 neuronas ocultas y una función de activación en la segunda capa del tipo Sigmoid.

Hiperparámetro	**Valor optimizado**
batch size	20
max epochs	100
hidden neurons	100
activation function	Sigmoid

Tabla 5.16: Valor de los hiperparámetros optimizados en la BNN para la discriminación de bacterias *E. Coli* resistentes y sensibles al Ceftriaxone.

Evaluación del modelo BNN

De las 88 muestras que forman el subconjunto de *test*, 36 resultan ser resistentes al Ceftriaxone y las 52 restante, sensibles. A partir de la matriz de confusión generada al evaluar el modelo de BNN con estas muestras (véase en la Tabla 5.17), se calculan los valores de las figuras de mérito obteniendo una *accuracy* del 63.31 %, una *precision* del 68.66 %, una *recall* del 88.46 %, una *specificity* del 41.66 %, una *balanced accuracy* del 65.06 % y un valor de área bajo la curva ROC de 0.651.

		Valor Clasificación	
		Sensible	Resistente
Valor Real	Sensible	46	6
	Resistente	21	15

Tabla 5.17: Matriz de confusión para discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ceftriaxone por la BNN.

Asimismo, la Figura 5.11 representa gráficamente una comparativa entre la clasificación real y la predicha por el modelo, mostrando de color amarillo las muestras sensibles y de color morado las resistentes al antibiótico.

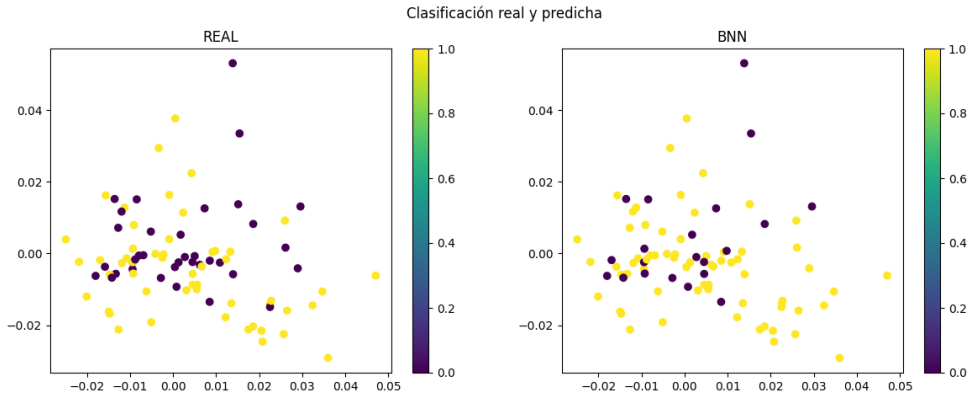


Figura 5.11: Representación gráfica en 2 dimensiones de la discriminación de bacterias *E. Coli* resistentes (de color morado) y sensibles (de color amarillo) al antibiótico Ceftriaxone.

Comparación con los algoritmos RF y LightGBM

Al igual que la BNN, en el entrenamiento de RF y LightGBM también se han optimizado sus hiperparámetros (véase en la Tabla 5.18 donde se representa a la izquierda los hiperparámetros de RF y a la derecha los de LightGBM). El modelo RF cuenta con 100 árboles de decisión, una profundidad de 30 nodos, un mínimo de 1 muestra por hoja, un máximo de 75 variables que tener considerar en cada árbol y un mínimo de 2 muestras para dividir el nodo. El algoritmo LightGBM se ha diseñado con 200 árboles, una tasa de aprendizaje de 0.1, una profundidad de 8 nodos y un número mínimo de 20 hojas por árbol.

Hiperparámetro	**Valor optimizado**
Estimators	100
Max Depth	30
Min Samples per leaf	1
Max Features	75
Min Split size	5

Hiperparámetro	**Valor optimizado**
Estimators	200
Learning rate	0.1
Number of leaves	8
Min samples per leaf	20

Tabla 5.18: Hiperparámetros optimizados en el entrenamiento de los algoritmos RF (izquierda) y LightGBM (derecha) para la discriminación de especies *E. Coli* resistentes y sensibles al antibiótico Ceftriaxone.

Al registrar las prestaciones obtenidas con los modelos RF, LightGBM y BNN se obtiene la Tabla 5.19, donde LightGBM alcanza mejores valores para la mayoría de prestaciones excepto para *precision* y *specificity*, lográndolo la BNN que categoriza la clase resistente con mayor acierto que el resto de modelos.

Algoritmo	***Accuracy***	***Precision***	***Recall***	***Specificity***	***Balanced Accuracy***	**AUC**
RF	61.36 %	58.33 %	90.38 %	19.44 %	54.91 %	0.693
LightGBM	**71.59 %**	68 %	**98.08 %**	33.33 %	**65.71 %**	**0.727**
BNN	63.31 %	**68.66 %**	88.46 %	**41.66 %**	65.06 %	0.651

Tabla 5.19: Comparación de las métricas obtenidas en los distintos algoritmos para la discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Ceftriaxone. Se muestran en negrita los mejores resultados.

5.4.5. Discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Cefepime

Finalmente, en esta última Subsección se pretende clasificar las muestras de la bacteria *E. Coli* atendiendo al tipo de respuesta (resistente o sensible) que presenta ante el antibiótico Cefepime.

Entrenamiento del modelo BNN con *Stratified* 5-*Fold CV*

Para el entrenamiento del modelo, se toma el subconjunto de las muestras de *train* de las que 31 son resistentes al Cefepime y las 138 resultantes, presentan una respuesta sensible al antibiótico. Con ellas se lleva a cabo la primera fase del diseño del modelo clasificatorio en la que se optimizan los hiperparámetros de interés, expuestos a continuación en la Tabla 5.20. Por ende, en el diseño de a BNN se ejecuta el algoritmo de retropropagación un máximo de 100 veces separando las muestras en 10 para cada iteración. Asimismo, se definen para la arquitectura de la red un número de 110 neuronas ocultas y la función Sigmoid se utiliza como función de activación de la última capa.

Hiperparámetro	**Valor optimizado**
batch size	10
max epochs	100
hidden neurons	110
activation function	Sigmoid

Tabla 5.20: Valor de los hiperparámetros optimizados para la discriminación de bacterias *E. Coli* resistentes y sensibles al Cefepime.

Evaluación del modelo BNN

En la segunda fase de la etapa de diseño del modelo, se evalúa. En este caso se toman 19 muestras resistentes al Cefepime, y 55 sensibles, que hacen un total de 74, y se obtiene la matriz de confusión representada en la Tabla 5.21. A partir de ella se calculan las figuras de mérito: *accuracy* del 77.03 %, *precision* del 76.38 %, *recall* del 100 %, *specificity* del 10.53 % y *balanced accuracy* del 55.26 %. Asimismo, se obtiene 0.50 como valor de AUC.

		Valor Clasificación	
		Sensible	Resistente
Valor Real	Sensible	55	0
	Resistente	17	2

Tabla 5.21: Matriz de confusión para discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Cefepime de la BNN.

En la Figura 5.12 se compara la clasificación real y la predicha por la BNN, de manera que las muestras de color morado son las resistentes y las de color amarillo son sensibles al Cefepime.

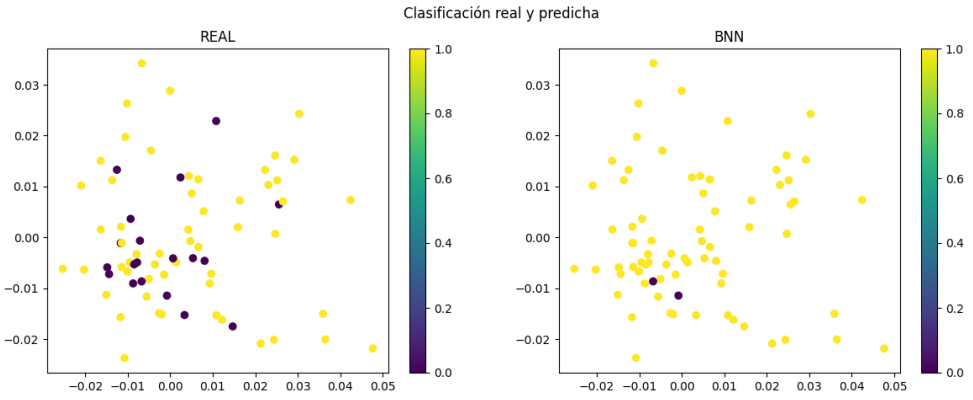


Figura 5.12: Representación gráfica en 2 dimensiones de la discriminación de bacterias *E. Coli* resistentes (de color morado) y sensibles (de color amarillo) al antibiótico Cefepime.

Comparación con los algoritmos RF y LightGBM

Los modelos RF y LightGBM se diseñan atendiendo a la combinación de hiperparámetros que obtienen el mayor valor de *balanced accuracy*, mostrándose en la Tabla 5.22. En primer lugar, RF ha sido diseñado con 50 árboles de decisión, una longitud máxima entre el nodo raíz

y una hoja de 10, con al menos 1 muestra por hoja y un número máximo de 12 variables y mínimo de 2 muestras a tener en cuenta en la división. Finalmente, el modelo LightGBM se ha implementado con 100 árboles, una tasa de 1 para minimizar el sobreajuste, una profundidad de 8 nodos hojas y un mínimo de 16 hojas por árbol.

Hiperparámetro	Valor optimizado
Estimators	50
Max Depth	10
Min Samples per leaf	1
Max Features	12
Min Split size	2

Hiperparámetro	Valor optimizado
Estimators	100
Learning rate	1
Number of leaves	8
Min samples per leaf	16

Tabla 5.22: Hiperparámetros optimizados en el entrenamiento de los algoritmos RF (izquierda) y LightGBM (derecha) respectivamente para la discriminación de especies *E. Coli* resistentes y sensibles al antibiótico Cefepime.

En la fase de evaluación de los modelos se obtienen unas figuras de mérito para comprobar su rendimiento. Estos valores se muestran en la Tabla 5.23 para los modelos RF, LightGBM y BNN. En la tabla, se observa cómo LightGBM y BNN han coincidido en la clasificación de las muestras, obteniendo las mismas prestaciones. Estos modelos son capaces de categorizar correctamente dos muestras de la clase resistente, sin embargo, RF no ha clasificado como resistente ninguna de las muestras, sino que clasifica todas como sensibles, obteniendo valores para sus figuras de mérito altos a pesar de no realizar un buen proceso de clasificación.

Algoritmo	*Accuracy*	*Precision*	*Recall*	*Specificity*	*Balanced Accuracy*	AUC
RF	74.32 %	74.32 %	**100 %**	0 %	50 %	**0.632**
LightGBM	**77.03 %**	**76.39 %**	**100 %**	**10.53 %**	**55.26 %**	0.572
BNN	**77.03 %**	**76.39 %**	**100 %**	**10.53 %**	**55.26 %**	0.572

Tabla 5.23: Comparación de las métricas obtenidas en los distintos algoritmos para la discriminación de bacterias *E. Coli* resistentes y sensibles al antibiótico Cefepime. Se muestran en negrita los mejores resultados.

Capítulo 6

Conclusiones y líneas futuras

En este capítulo se presentan las conclusiones obtenidas tras la realización del presente TFG y además se incluyen posibles líneas futuras de trabajo que se han identificado con el fin de continuar este estudio.

6.1. Conclusiones

Para la realización de este TFG se han tomado los espectros de masas obtenidos con la técnica MALDI-TOF MS a partir de muestras del hongo *Aspergillus* y de la bacteria *E. Coli*, aportados por Clover Biosoft.

En primer lugar, ambos conjuntos se preprocesaron con el objetivo de obtener unos espectros de masas limpios y las intensidades de los picos queden reflejadas en una matriz de picos, para su posterior procesado.

Tras realizar una separación de los conjuntos en subconjuntos de *train* y *test* en un proporción del 70/30 respectivamente. Además, se lleva a cabo una detección de valores atípicos con dos métodos diferentes que no coinciden en sus resultados, por lo que se toma el criterio de no despreciar muestras para el entrenamiento del modelo de ML.

A continuación, se ejecuta la reducción de dimensiones de las características en ambos conjuntos de datos con PCA para que el entrenamiento del modelo de BNN sea eficiente. Asimismo, el modelo es evaluado con el subconjunto de muestras de *test* y los resultados de las figuras de mérito se comparan con aquellos obtenidos al realizar los mismos procesos de clasificación con los algoritmos de ML RF y LightGBM.

En un enfoque general, en el conjunto *A. fumigatus*, la cantidad de muestras perteneciente a cada clase es menor que en el conjunto de *E. Coli*, debido tratarse de un conjunto con menores dimensiones. Además, ambos conjuntos presentan un gran desbalanceo en sus clases, destacando principalmente la discriminación entre la especie *A. fumigatus s.s.* y las especies crípticas del primer conjunto (106 frente a 17 en el subconjunto de *train*) y la discriminación entre las muestras de *E. Coli* con respuesta resistente y sensible al Cefepime en el segundo conjunto (31 frente a 138 en el subconjunto de *train*). Como estrategias para tratar este desbalanceo se utiliza la técnica *Stratified 5-Fold CV* en el entrenamiento de la BNN y en la optimización de los hiperparámetros se utiliza la métrica *balanced accuracy* que tiene en cuenta ambas clases. Se comprueba que este desbalanceo de clases interviene claramente en el rendimiento del modelo ya que, de las discriminaciones efectuadas en cada conjunto, las citadas anteriormente como aquellas con mayor desbalanceo, son en las que los tres clasificadores obtienen valores inferiores para las prestaciones *balanced accuracy* y *AUC*, comparándolos con los resultados de las diferentes discriminaciones de cada conjunto.

Adicionalmente, los resultados obtenidos en el primer conjunto son bastante mejores que en el segundo, siendo la BNN el modelo que obtiene el valor más alto para todas las prestaciones tanto para la discriminación entre la especie *A. fumigatus s.s.* y las especies crípticas, como para la clasificación de la especie *A. fumigatus s.s.* en individuos sensibles y resistentes a los azoles. Sin embargo, en las discriminaciones de las muestras de *E. Coli* en sensibles y resistentes a los antibióticos Ciprofloxacin, Ceftriaxone y Cefepime la selección del mejor modelo no es tan clara. Para las muestras con relación a los antibióticos Ciprofloxacin y Ceftriaxone, el modelo LightGBM obtiene valores para *balanced accuracy* y AUC mejores al resto de modelos, aunque la BNN presenta resultados superiores en *precision* y *specificity*, clasificando la clase resistente con mayor acierto que el resto de modelos. Sin embargo, en el caso de discriminación entre muestras de *E. Coli* resistentes y sensibles al Cefepime, los modelos LightGBM y BNN coinciden en todos los resultados debido a haber clasificado correctamente la misma proporción de muestras resistentes y sensibles, aunque el modelo RF obtiene un valor de AUC superior debido a que categoriza como sensibles todas las muestras, aún habiendo muestras resistentes. Por lo tanto, el modelo RF es claramente el que tiene un rendimiento peor en todas las discriminaciones realizadas en ambos conjuntos de datos.

No obstante, los resultados en las clasificaciones asociadas al conjunto de *E. Coli*, a pesar de tener porcentajes altos en prestaciones como *accuracy, precision, recall* o AUC, no son adecuados puesto que los modelos consideran la gran mayoría de las muestras como la clase predominante (sensible), resultando sencillo el acierto debido a la pequeña cantidad de muestras en la clase minoritaria (resistente). Por lo que en este caso, no se pueden considerar modelos

realmente adecuados en la clasificación de este conjunto de muestras de la especie *E. Coli*, ya que van a definir muestras resistentes como sensibles al antibiótico, suponiendo un error que no va a ayudar a paliar la RAM.

En definitiva, el algoritmo BNN se trata de un modelo de ML fiable capaz de clasificar distintas clases de muestras con un rendimiento, en general, superior a los que están incluidos en la plataforma de Clover, obteniendo mejores prestaciones que RF para las cinco distinciones realizadas y que LightGBM en las clasificaciones correspondientes al conjunto *Aspergillus*. Esto convierte a la BNN en un buen candidato para ser incluido en la plataforma.

Estos resultados sirven para argumentar que la combinación de la técnica MALDI-TOF MS y el ML es competente para participar en la lucha e la resistencia antimicrobiana facilitando a los clínicos y a la industria farmacéutica información útil en su propósito.

6.2. Líneas futuras

El presente TFG supone una pequeña aportación a la combinación entre las técnicas de ML y el análisis de datos clínicos, en este caso de de resistencias antimicrobianas. Sin embargo, este TFG puede ser de utilidad hacia otras investigaciones que puedan ir progresando el trabajo realizado. Entre posibles líneas futuras se encuentran:

- Añadir técnicas para tratar el desbalanceo de las clases y estudiar cómo se comporta el modelo de BNN.
- Considerar como valores atípicos todas las muestras que consideran como tal las técnicas de reconstrucción PCA y correlación espectral para comparar los resultados obtenidos.
- Utilizar más muestras de espectros de masas obtenidos con MALDI-TOF MS para seguir validando y mejorando el modelo de BNN, recurriendo también a otros microorganismos como virus y protozoos.
- Comprobar si el rendimiento de la BNN aumenta al variar la estructura de la red, modificando las capas bayesianas lineales por capas bayesianas convolucionales, combinándolas, o incluso utilizando distintas cantidades de capas.
- Aumentar la velocidad en la ejecución del modelo con un procesamiento GPU (del inglés *Graphics Processing Unit*), pudiendo así ampliar la cantidad de muestras de entrada.
- Extrapolar este estudio a espectros resultantes de la espectroscopía FTIR.

Bibliografía

[1] Antony Croxatto, Guy Prod'hom, and Gilbert Greub. Applications of maldi-tof mass spectrometry in clinical diagnostic microbiology. *FEMS microbiology reviews*, 36(2):380–407, 2012.

[2] CLOVER Bioanalytical Software S.L. Clover ms data analysis software, 2015. Última consulta a 2 de julio de 2023. Disponible en https://www.clovermsdataanalysis.com/.

[3] scikit-learn developers. Cross-validation: evaluating estimator performance, 2007 - 2023. Última consulta a 15 de junio de 2023. Disponible en https://scikit-learn.org/stable/modules/cross_validation.html.

[4] Johnson Kolluri, Vinay Kumar Kotte, MSB Phridviraj, and Shaik Razia. Reducing overfitting problem in machine learning using novel l1/4 regularization method. In *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pages 934–938. IEEE, 2020.

[5] Andrej Krenker, Janez Bešter, and Andrej Kos. Introduction to the artificial neural networks. *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech*, pages 1–18, 2011.

[6] scikit-learn developers. Decision trees, 2007 - 2023. Última consulta a 15 de junio de 2023. Disponible en https:https://scikit-learn.org/stable/modules/tree.html#classification.

[7] Di-ni Wang, Lang Li, and Da Zhao. Corporate finance risk prediction based on lightgbm. *Information Sciences*, 602:259–268, 2022.

[8] Ventola C Lee. The antibiotic resistance crisis: part 1: causes and threats. *Pharmacy and therapeutics*, 40(4):277, 2015.

[9] Xiao-Yang Hu, Martin Logue, and Nicola Robinson. Antimicrobial resistance is a global problem–a uk perspective. *European journal of integrative medicine*, 36:101136, 2020.

[10] Juan-Ignacio Alós. Resistencia bacteriana a los antibióticos: una crisis global. *Enfermedades infecciosas y microbiología clínica*, 33(10):692–699, 2015.

[11] Cristina Eugenia Cabrera, Rommel Fabián Gómez, and Andrés Edmundo Zúñiga. La resistencia de bacterias a antibióticos, antisépticos y desinfectantes una manifestación de los mecanismos de supervivencia y adaptación. *Colombia médica*, 38(2):149–158, 2007.

[12] Ventola C Lee. The antibiotic resistance crisis: part 2: management strategies and new agents. *Pharmacy and Therapeutics*, 40(5):344, 2015.

[13] Sibhghatulla Shaikh, Jamale Fatima, Shazi Shakil, Syed Mohd Danish Rizvi, and Mohammad Amjad Kamal. Antibiotic resistance and extended spectrum beta-lactamases: Types, epidemiology and treatment. *Saudi journal of biological sciences*, 22(1):90–101, 2015.

[14] Andrew Amato-Gauci, Andrea Ammon, et al. The first european communicable disease epidemiological report. *European Centre for Disease Prevention and Control, Stockholm*, 1:1–360, 2007.

[15] Laura Freire-Moran, Bo Aronsson, Chris Manz, Inge C Gyssens, Anthony D So, Dominique L Monnet, Otto Cars, ECDC-EMA working group, et al. Critical shortage of new antibiotics in development against multidrug-resistant bacteria—time to react is now. *Drug resistance updates*, 14(2):118–124, 2011.

[16] Jim O'Neill. Tackling drug-resistant infections globally: final report and recommendations. 2016.

[17] Porooshat Dadgostar. Antimicrobial resistance: implications and costs. *Infection and drug resistance*, pages 3903–3910, 2019.

[18] Asamblea Mundial de la Salud. Plan de acción mundial sobre la resistencia a los antimicrobianos: Informe de la secretaría. Technical report, Organización Mundial de la Salud, 2016.

[19] Caroline Weis, Aline Cuénod, Bastian Rieck, Olivier Dubuis, Susanne Graf, Claudia Lang, Michael Oberle, Maximilian Brackmann, Kirstine K Søgaard, Michael Osthoff, et al. Direct antimicrobial resistance prediction from clinical maldi-tof mass spectra using machine learning. *Nature Medicine*, 28(1):164–174, 2022.

[20] CLOVER Bioanalytical Software S.L. Clover biosoft, 2015. Última consulta a 20 de junio de 2023. Disponible en https://cloverbiosoft.com/.

[21] Trevor John Franklin and George Alan Snow. *Biochemistry of antimicrobial action*. Springer, 2013.

[22] Dianelys Quiñones Pérez. Resistencia antimicrobiana: evolución y perspectivas actuales ante el enfoqueüna salud". *Revista Cubana de Medicina Tropical*, 69(3):1–17, 2017.

[23] Edgar Ortiz Brizuela, Alberto Ordinola Navarro, and Bruno Ali López Luis. ¿ un mundo sin antibióticos? conoce la resistencia antimicrobiana. *Revista Digital Universitaria*, 24(3), 2023.

[24] Gang Liu, Line Elnif Thomsen, and John Elmerdahl Olsen. Antimicrobial-induced horizontal transfer of antimicrobial resistance genes in bacteria: A mini-review. *Journal of Antimicrobial Chemotherapy*, 77(3):556–567, 2022.

[25] Ivone Alexandra GARCÍA SALAZAR. *Aislamiento de bacteriófagos de Pseudomonas aeruginosa multidrogo–resistente en aguas de tres ríos de la Provincia de Lima-Perú*. PhD thesis, Universidad Nacional Mayor de San Marcos, 2018.

[26] World Health Organization et al. Global action plan on antimicrobial resistance. 2015.

[27] Chilot Abiyu Demeke, Getnet Mequanent Adinew, Tamrat Befekadu Abebe, Abebech Tewabe Gelaye, Sisay G/Hana Gemeda, and Dawit Kumilachew Yimenu. Comparative analysis of the effectiveness of narrow-spectrum versus broad-spectrum antibiotics for the treatment of childhood pneumonia. *SAGE open medicine*, 9:20503121211044379, 2021.

[28] Etienne Carbonnelle and L Raskine. Maldi-tof mass spectrometry tools for bacterial identification in clinical microbiology laboratory. *Bio tribune magazine*, 39:35–42, 2011.

[29] Robin Patel. Maldi-tof ms for the diagnosis of infectious diseases. *Clinical chemistry*, 61(1):100–111, 2015.

[30] Marina Oviaño and Belén Rodríguez-Sánchez. Maldi-tof mass spectrometry in the 21st century clinical microbiology laboratory. *Enfermedades infecciosas y microbiologia clinica (English ed.)*, 39(4):192–200, 2021.

[31] Margarita Estreya Zvezdanova, Manuel J Arroyo, Gema Méndez, Ana Candela, Luis Mancera, Julio García Rodríguez, Julia Lozano Serra, Rosa Jiménez, Inmaculada Lozano, Carmen Castro, et al. Detection of azole resistance in aspergillus fumigatus complex isolates

using maldi-tof mass spectrometry. *Clinical Microbiology and Infection*, 28(2):260–266, 2022.

[32] Ana Candela, Manuel J Arroyo, Ángela Sánchez-Molleda, Gema Méndez, Lidia Quiroga, Adrián Ruiz, Emilia Cercenado, Mercedes Marín, Patricia Muñoz, Luis Mancera, et al. Rapid and reproducible maldi-tof-based method for the detection of vancomycin-resistant enterococcus faecium using classifying algorithms. *Diagnostics*, 12(2):328, 2022.

[33] Frédéric Lamoth. Aspergillus fumigatus-related species in clinical practice. *Frontiers in microbiology*, 7:683, 2016.

[34] Pilar Escribano, Belén Rodríguez-Sánchez, Judith Díaz-García, María Teresa Martín-Gómez, Elisa Ibáñez-Martínez, María Rodríguez-Mayo, Teresa Peláez, Elia García-Gómez de la Pedrosa, Rocío Tejero-García, José María Marimón, et al. Azole resistance survey on clinical aspergillus fumigatus isolates in spain. *Clinical Microbiology and Infection*, 27(8):1170–e1, 2021.

[35] Margarita Estreya Nikolaeva Zvezdanova. Identificación de hongos filamentosos y levaduras de interés clínico mediante espectrometría de masas maldi-tof. 2022.

[36] Julia Serrano-Lobo, Ana Gómez, Belén Rodríguez-Sánchez, Patricia Muñoz, Pilar Escribano, and Jesús Guinea. Azole-resistant aspergillus fumigatus clinical isolate screening in azole-containing agar plates (eucast e. def 10.1): low impact of plastic trays used and poor performance in cryptic species. *Antimicrobial Agents and Chemotherapy*, 65(8):e00482–21, 2021.

[37] Nerino Allocati, Michele Masulli, Mikhail F Alexeyev, and Carmine Di Ilio. Escherichia coli in europe: an overview. *International journal of environmental research and public health*, 10(12):6235–6254, 2013.

[38] Araceli Vázquez-Rojas and Reyna Miliar-De Jesús. Enfermedades. *Revista de Enfermedades Infecciosas en Pediatría*, 33(133):1713, 2020.

[39] Ana Betrán, María José Lavilla, Rocío Cebollada, José Manuel Calderón, and Luís Torres. Resistencia antibiótica de escherichia coli en infecciones urinarias nosocomiales y adquiridas en la comunidad del sector sanitario de huesca 2016-2018. *Revista Clínica de Medicina de Familia*, 13(3):198–202, 2020.

[40] Laurent Poirel, Jean-Yves Madec, Agnese Lupo, Anne-Kathrin Schink, Nicolas Kieffer, Patrice Nordmann, and Stefan Schwarz. Antimicrobial resistance in escherichia coli. *Microbiology Spectrum*, 6(4):6–4, 2018.

[41] Armand Paauw, Debby Jonker, Guus Roeselers, Jonathan ME Heng, Roos H Mars-Groenendijk, Hein Trip, E Margo Molhoek, Hugo-Jan Jansen, Jan van der Plas, Ad L de Jong, et al. Rapid and reliable discrimination between shigella species and escherichia coli using maldi-tof mass spectrometry. *International Journal of Medical Microbiology*, 305(4-5):446–452, 2015.

[42] Clifford G Clark, Peter Kruczkiewicz, Cai Guan, Stuart J McCorrister, Patrick Chong, John Wylie, Paul van Caeseele, Helen A Tabor, Phillip Snarr, Matthew W Gilmour, et al. Evaluation of maldi-tof mass spectroscopy methods for determination of escherichia coli pathotypes. *Journal of microbiological methods*, 94(3):180–191, 2013.

[43] Catherine Berthomieu and Rainer Hienerwadel. Fourier transform infrared (ftir) spectroscopy. *Photosynthesis research*, 101:157–170, 2009.

[44] Ahmed Fadlelmoula, Susana O Catarino, Graça Minas, and Vítor Carvalho. A review of machine learning methods recently applied to ftir spectroscopy data for the analysis of human blood cells. *Micromachines*, 14(6):1145, 2023.

[45] Belén Rodríguez-Sánchez, Ana Candela, Manuel J Arroyo, María Sánchez-Cueto, Mercedes Marín, Emilia Cercenado, Gema Méndez, Patricia Muñoz, Luis Mancera, and David Rodriguez-Temporal. Rapid discrimination of pseudomonas aeruginosa st175 isolates involved in a nosocomial outbreak using maldi-tof mass spectrometry and ftir spectroscopy coupled with machine learning. *Authorea Preprints*, 2022.

[46] Batta Mahesh. Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9:381–386, 2020.

[47] Qifang Bi, Katherine E Goodman, Joshua Kaminsky, and Justin Lessler. What is machine learning? a primer for the epidemiologist. *American journal of epidemiology*, 188(12):2222–2239, 2019.

[48] Peter Harrington. *Machine learning in action*. Simon and Schuster, 2012.

[49] Arunim Garg and Vijay Mago. Role of machine learning in medical research: A survey. *Computer science review*, 40:100370, 2021.

[50] Hugo López-Fernández, Hugo Miguel Santos, José L Capelo, Florentino Fdez-Riverola, Daniel Glez-Peña, and Miguel Reboiro-Jato. Mass-up: an all-in-one open software application for maldi-tof mass spectrometry knowledge discovery. *BMC bioinformatics*, 16:1–12, 2015.

[51] Eugenio Del Prete, Diego d'Esposito, Maria Fiorella Mazzeo, Rosa Anna Siciliano, and Angelo Facchiano. Comparative analysis of maldi-tof mass spectrometric data in proteomics: a case study. In *Computational Intelligence Methods for Bioinformatics and Biostatistics: 12th International Meeting, CIBB 2015, Naples, Italy, September 10-12, 2015, Revised Selected Papers 12*, pages 154–164. Springer, 2016.

[52] Eugenio Del Prete, Angelo Facchiano, Aldo Profumo, Claudia Angelini, and Paolo Romano. Geenar: A web tool for reproducible maldi-tof analysis. *Frontiers in Genetics*, 12:635814, 2021.

[53] Fan Mo, Qun Mo, Yuanyuan Chen, David R Goodlett, Leroy Hood, Gilbert S Omenn, Song Li, and Biaoyang Lin. Waveletquant, an improved quantification software based on wavelet signal threshold de-noising for labeled quantitative proteomic analysis. *BMC bioinformatics*, 11:1–9, 2010.

[54] Anne C Sauve and Terence P Speed. Normalization, baseline correction and alignment of high-throughput mass spectrometry data. *Proceedings Gensips*, pages 1–4, 2004.

[55] Anestis Antoniadis, Jérémie Bigot, and Sophie Lambert-Lacroix. Peaks detection and alignment for mass spectrometry data. *Journal de la Société Française de Statistique*, 151(1):17–37, 2010.

[56] Sören-Oliver Deininger, Dale S Cornett, Rainer Paape, Michael Becker, Charles Pineau, Sandra Rauser, Axel Walch, and Eryk Wolski. Normalization in maldi-tof imaging datasets of proteins: practical considerations. *Analytical and bioanalytical chemistry*, 401:167–181, 2011.

[57] Solveig Badillo, Balazs Banfai, Fabian Birzele, Iakov I Davydov, Lucy Hutchinson, Tony Kam-Thong, Juliane Siebourg-Polster, Bernhard Steiert, and Jitao David Zhang. An introduction to machine learning. *Clinical pharmacology & therapeutics*, 107(4):871–885, 2020.

[58] Jose J Salazar, Lean Garland, Jesus Ochoa, and Michael J Pyrcz. Fair train-test split in machine learning: Mitigating spatial autocorrelation for improved prediction accuracy. *Journal of Petroleum Science and Engineering*, 209:109885, 2022.

[59] Gang Luo. A review of automatic selection methods for machine learning algorithms and hyper-parameter values. *Network Modeling Analysis in Health Informatics and Bioinformatics*, 5:1–16, 2016.

[60] Sebastian Raschka, Yuxi Hayden Liu, Vahid Mirjalili, and Dmytro Dzhulgakov. *Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python*. Packt Publishing Ltd, 2022.

[61] Joseph V Roshan. Optimal ratio for data splitting. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 15(4):531–538, 2022.

[62] Sotiris B Kotsiantis, Dimitris Kanellopoulos, and Panagiotis E Pintelas. Data preprocessing for supervised leaning. *International journal of computer science*, 1(2):111–117, 2006.

[63] Cao LJ, Kok Seng Chua, WK Chong, HP Lee, and QM Gu. A comparison of pca, kpca and ica for dimensionality reduction in support vector machine. *Neurocomputing*, 55(1-2):321–336, 2003.

[64] Hooman H Rashidi, Nam K Tran, Elham Vali Betts, Lydia P Howell, and Ralph Green. Artificial intelligence and machine learning in pathology: the present landscape of supervised methods. *Academic pathology*, 6:2374289519873088, 2019.

[65] Neha Gupta et al. Artificial neural network. *Network and Complex Systems*, 3(1):24–28, 2013.

[66] Maritza Correa, Concha Bielza, and J Pamies-Teixeira. Comparison of bayesian networks and artificial neural networks for quality detection in a machining process. *Expert systems with applications*, 36(3):7270–7279, 2009.

[67] Leke Zajmi, Falah YH Ahmed, and Adam Amril Jaharadak. Concepts, methods, and performances of particle swarm optimization, backpropagation, and neural networks. *Applied Computational Intelligence and Soft Computing*, 2018, 2018.

[68] Arangio S and Beck JL. Bayesian neural networks for bridge integrity assessment. *Structural Control and Health Monitoring*, 19(1):3–21, 2012.

[69] Lesley Ofelia Mesa Páez, Miller Rivera Lozano, and Jesús Andrés Romero Davila. Descripción general de la inferencia bayesiana y sus aplicaciones en los procesos de gestión. *La simulación al Servicio de la Academia*, 2:1–28, 2011.

[70] Vikram Mullachery, Aniruddh Khera, and Amir Husain. Bayesian neural networks. *arXiv preprint arXiv:1801.07710*, 2018.

[71] Javier Rodríguez Mañanes et al. Redes bayesianas y redes neuronales como modelos del aprendizaje casual. 2009.

[72] Vincent Fortuin, Adrià Garriga-Alonso, Mark van der Wilk, and Laurence Aitchison. Bnn-priors: A library for bayesian neural network inference with different prior distributions. *Software Impacts*, 9:100079, 2021.

[73] Shaojie Jiang, Pengjie Ren, Christof Monz, and Maarten de Rijke. Improving neural response diversity with frequency-aware cross-entropy loss. In *The World Wide Web Conference*, pages 2879–2885, 2019.

[74] James M Joyce. Kullback-leibler divergence. In *International encyclopedia of statistical science*, pages 720–722. Springer, 2011.

[75] Sebastian Bock, Josef Goppold, and Martin Weiß. An improvement of the convergence proof of the adam-optimizer. *arXiv preprint arXiv:1804.10587*, 2018.

[76] Himani Sharma, Sunil Kumar, et al. A survey on decision tree algorithms of classification in data mining. *International Journal of Science and Research (IJSR)*, 5(4):2094–2097, 2016.

[77] Victor Francisco Rodriguez-Galiano, Bardan Ghimire, John Rogan, Mario Chica-Olmo, and Juan Pedro Rigol-Sanchez. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS journal of photogrammetry and remote sensing*, 67:93–104, 2012.

[78] Luay Fraiwan, Khaldon Lweesy, Natheer Khasawneh, Heinrich Wenz, and Hartmut Dickhaus. Automated sleep stage identification system based on time–frequency analysis of a single eeg channel and random forest classifier. *Computer methods and programs in biomedicine*, 108(1):10–19, 2012.

[79] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30:11.

[80] Yan Wang and Tao Wang. Application of improved lightgbm model in blood glucose prediction. *Applied Sciences*, 10(9):3227, 2020.

[81] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*, 2020.

[82] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240, 2006.

[83] Microsoft. Visual studio code, 2023. Última consulta a 11 de julio de 2023. Disponible en https://code.visualstudio.com/.

[84] Daza Pérez RM. Resistencia bacteriana a antimicrobianos: su importancia en la toma de decisiones en la práctica diaria. *Inf Ter Sist Nac Salud*, 22:57–67, 1998.

[85] Sagar Imambi, Kolla Bhanu Prakash, and GR Kanagachidambaresan. Pytorch. *Programming with TensorFlow: Solution for Edge Computing Applications*, pages 87–104, 2021.

[86] Fatih Ertam and Galip Aydın. Data classification with deep learning using tensorflow. In *2017 international conference on computer science and engineering (UBMK)*, pages 755–758. IEEE, 2017.

[87] Antonio Gulli and Sujit Pal. *Deep learning with Keras*. Packt Publishing Ltd, 2017.

[88] Benjamin Bossan Marian Tietz, Daniel Nouri. Skorch documentation, 2017. Última consulta a 4 de julio de 2023. Disponible en https://skorch.readthedocs.io/en/stable/.

[89] scikit-learn developers. scikit-learn machine learning in python, 2007 - 2023. Última consulta a 4 de julio de 2023. Disponible en https://scikit-learn.org/stable/index.html.

[90] scikit-learn developer. Random forest classifier, 2007-2023. Última consulta a 11 de julio de 2023. Disponible en https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

[91] Microsoft Corporation. Lightgbm classifier, 2023. Última consulta a 11 de julio de 2023. Disponible en https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMClassifier.html.