UNIVERSIDAD AUTÓNOMA DE SINALOA

FACULTAD DE INFORMÁTICA

DOCTORADO EN CIENCIAS DE LA INFORMACIÓN

Metodología experimental para la selección de variables en un problema de búsqueda de decaimientos en el experimento Belle II, utilizando algoritmos de aprendizaje automático

TESIS

QUE COMO REQUISITO PARA OBTENER EL GRADO DE DOCTOR EN CIENCIAS DE LA INFORMACIÓN

PRESENTA:

M.C. DAVID RODRÍGUEZ PÉREZ

DIRECTORES DE TESIS:

DR. PEDRO LUIS MANUEL PODESTA LERMA

DRA. ISABEL DOMÍNGUEZ JIMÉNEZ

Culiacán, Sinaloa. 24 de enero de 2024

Dedicado a

mi familia

Agradecimientos

Quisiera expresar mi profundo agradecimiento a todas las personas que contribuyeron de manera significativa a la realización de esta tesis. Este proyecto no habría sido posible sin el apoyo y la guía de personas maravillosas.

En primer lugar, quiero agradecer a mis asesores, Dra. Isabel Domínguez Jiménez, Dr. Pedro L. M. Podesta Lerma y Dr. Inés Fernando Vega López cuya experiencia y dedicación fueron fundamentales para dar forma a este trabajo. Sus valiosos consejos fueron de gran ayuda en este viaje académico, eso sin mencionar la paciencia que tuvieron conmigo.

Agradezco también a mis profesores y profesoras por impartir conocimientos que formaron la base de este trabajo, en especial a la Dra. Xiomara Penélope Zaldívar Colado, Dr. Jorge Adalberto Navarro Castillo y Dr. Arturo Yee Rendón. Cada lección aprendida en el aula contribuyó al desarrollo de este proyecto de investigación.

Por último, quiero agradecer a la Universidad Autónoma de Sinaloa, a la Facultad de Informática Culiacán, al Posgrado en Ciencias de la Información y al Consejo Nacional de Ciencia y Tecnología por permitirme culminar esta etapa de mi vida.

The authors would like to thank the financial support provided by the Consejo Nacional de Ciencia y Tecnología (CONACyT) with grant number A1-S-33202 and Universidad Autonoma de Sinaloa (UAS) with grant numbers PRO-A8-048, PRO-A1-018

Resumen

El uso de algoritmos inteligentes en la búsqueda de decaimientos de partículas en experimentos de altas energías representa una valiosa oportunidad tanto para las ciencias de la información como para la física de altas energías. Estas técnicas tienen el potencial de mejorar la clasificación de diversos decaimientos en este contexto.

Si bien el aprendizaje automático ya ha demostrado su utilidad en la física de altas energías, con mejoras significativas en diferentes áreas y experimentos, cada proyecto experimental tiene requisitos específicos para alcanzar sus objetivos. Esta tesis se enfoca en la clasificación del decaimiento del leptón tau en el experimento Belle II. El objetivo es probar diversos algoritmos de inteligencia artificial para clasificar un decaimiento en particular: el decaimiento $\tau \to \pi^+\mu^-\mu^-\nu_\tau$, el cual aún no ha sido observado. Además, la metodología desarrollada puede ser extendida para considerar otros decaimientos que compartan características similares, es decir, aquellos que no presenten cambios en el número de partículas, sino en su tipo.

El experimento Belle II, conocido como una "fábrica de mesones B", produce una cantidad considerable de leptones tau. Durante el proceso de reconstrucción, se disponen de 162 variables relacionadas con el decaimiento en estudio. Este punto de encuentro entre las ciencias de la información y la física de altas energías es crucial, ya que la creación del conjunto de datos de entrada para los algoritmos de clasificación depende del decaimiento en estudio.

El primer desafío al que nos enfrentamos es determinar si todas las variables del conjunto inicial son relevantes para los algoritmos de clasificación, o si es posible descartar algunas de ellas para evitar la conocida "maldición de la dimensión". Para ello, se desarrolla una metodología de selección de variables que combina dos enfoques: el filtrado y la envoltura. El resultado es una metodología híbrida que extrae las mejores características de ambos enfoques.

Posteriormente, se realiza un breve análisis del origen de los datos para seleccionar los algoritmos de clasificación más apropiados. Sin embargo, esto no limita la posibilidad de aplicar otros algoritmos. Los algoritmos seleccionados incluyen regresión logística, árboles de

decisión, máquinas de vectores de soporte y perceptrón multicapa.

Una vez que se han determinado las variables relevantes y la técnica de aprendizaje automático adecuada, el objetivo en el problema de clasificación binaria es maximizar su rendimiento mediante el ajuste de los hiperparámetros correspondientes y la aplicación de las métricas de evaluación apropiadas para el problema en cuestión.

Por último, mencionamos dos propuestas que pueden ser implementadas en el experimento Belle II con el objetivo de ser consideradas por la colaboración para una posible incorporación en los análisis de decaimientos futuros.

Propuestas que toman en cuenta las principales aportaciones del trabajo, las cuales se basan en una selección de variables por medio de un método híbrido que permite reducir el número de clasificadores a construir y establecer el rendimiento para determinar un buen clasificador.

Abstract

The use of intelligent algorithms in the search for particle decays in high-energy experiments represents a valuable opportunity for both information sciences and high-energy physics. These techniques have the potential to improve the classification of various decays in this context.

This thesis, it focus on the classification of the τ lepton decay in the Belle II experiment. Our goal is to test various artificial intelligence algorithms to classify a particular decay: the decay $\tau \to \pi^+\mu^-\mu^-\nu_\tau$, which has not been observed yet. Furthermore, the developed methodology can be extended to consider other decays that share similar characteristics, i.e., those that do not involve changes in the number of particles but in their types.

The Belle II experiment, known as a "B meson factory,"produces a considerable amount of tau leptons. During the reconstruction process, we have access to 157 variables related to the decay under study. This intersection between information science and high-energy physics is crucial since the creation of the input dataset for the classification algorithms depends on the decay under study.

The first challenge is to determine if all variables in the initial dataset are relevant for the classification algorithms or if it is possible to discard some of them to avoid the well-known "curse of dimensionality."To do so, a variable selection methodology is developed to combine two approaches: filtering and wrapper methods. The result is a hybrid methodology that extracts the best features from both approaches.

Subsequently, it is conducted a brief analysis of the data to select the most suitable classification algorithms. However, this does not limit the possibility of applying other algorithms. The selected algorithms include logistic regression, decision trees, support vector machines, and multilayer perceptron.

Once the relevant variables and the appropriate machine learning technique have been determined, the objective in the binary classification problem is to maximize its performance by adjusting the corresponding hyperparameters and applying appropriate evaluation metrics

for the specific problem.

Finally, two proposals are mentioned, they could be implemented in the Belle II experiment with the aim of being considered by the collaboration for potential incorporation into future decay analysis.

Proposals that take into account the main contributions of the work, which are based on a variable selection through a hybrid method that allows reducing the number of classifiers to build and establish the performance to determine a good classifier.

Índice general

Índice de figuras

Capítulo 1

Introducción

Las ciencias de la información [Stock and Stock, 2013] son una ciencia joven en comparación con las matemáticas y la física, buena parte de su raíces se encuentran en ellas, por lo cual existe una relación intrínseca entre ellas, en los últimos años se ha incrementado la capacidad de acumular y estudiar conjunto de datos, que nunca hubieran soñado Newton o Gauss, Una de las primeras partes donde se tuvo está ingente cantidad de datos fue en el área de física de altas energías, en particular fueron los colisionadores de partículas, por ejemplo la red informática mundial (WWW, por sus siglas en ingles) [Berners-Lee et al., 2000] fue desarrollada en la Organización Europea para la Investigación Nuclear (CERN, por sus siglas en inglés) [Brüning et al., 2004] para poder comunicarse entre diferentes tipos de computadoras. Las ciencias de información han avanzado de manera vertiginosa y ahora son ellas las que están impulsando el desarrollo de la sociedad, y en particular la física de altas energías, ya que son muchas las técnicas que se usan, como son el reconocimiento de patrones [Bishop, 2006], la programación en paralelo [Almasi and Gottlieb, 1989], manejo de base de datos [Ullman and Widom, 2001], inteligencia artificial en partículas [Gupta et al., 2022, HEP ML Community,] el aprendizaje automático sobre el que nos enfocaremos en este trabajo.

El experimento Belle II [Kou et al., 2018], conocido como una "fábrica de mesones B", produce una cantidad considerable de leptones tau (del orden de 100 pares por segundo

y se esperan alrededor de 45 mil millones de pares de taus durante el funcionamiento del experimento que será de más de una década). Condiciones que serán descritas con más detalle en el Capítulo 4, pero que pueden ser resumidas de la siguiente manera: un colisionador de partículas, como su nombre lo indica colisiona partículas con el fin de observar el producto de dicha colisión, la cual a diferencia de una colisión en la vida cotidiana no resulta en fragmentos de las partículas colisionadas, ya que estas generalmente son partículas fundamentales y por lo tanto no se pueden fragmentar (Capítulo 2), en lugar de ello existe una transformación a otras partículas que pueden o no ser estables, de tal manera de que si no son estables vuelve a ocurrir otra transformación, esto hasta que solo existan partículas estables, dichas transformaciones son llamadas decaimientos. Siendo el propósito de los colisionadores y detectores determinar la historia de los decaimientos pero en el sentido inverso a la colisión, es decir, el análisis comienza con las partículas estables, después con las transformaciones o decaimientos intermedios y finalmente con las partículas que participaron en la colisión, este proceso es llamado reconstrucción. Esta reconstrucción se realiza sin conocer si las partículas estables o transformaciones intermedias fueron determinadas correctamente en su totalidad (esta fuera de la capacidad de los detectores), pero dado que se conocen las condiciones iniciales, es decir, las partículas que participan en la colisión, es posible determinar mediante estadística si la reconstrucción se realizó correctamente. Siendo este proceso de reconstrucción un área de oportunidad para las técnicas de aprendizaje automático de clasificación, ya que una reconstrucción solo puede ser buena o mala. Este es el punto de encuentro entre las ciencias de la información y la física de altas energías, ya que la creación del conjunto de datos de entrada para los algoritmos de clasificación depende del decaimiento en estudio.

Durante el proceso de reconstrucción se disponen de 162 variables relacionadas con el decaimiento en estudio y representa el primer desafío, ya que es necesario determinar si todas las variables del conjunto inicial son relevantes para los algoritmos de clasificación, o si es posible descartar algunas de ellas para evitar la conocida “maldición de la dimensión”. Para ello, se desarrolla una metodología de selección de variables que combina dos enfoques:

el filtrado y la envoltura. El resultado es una metodología híbrida que extrae las mejores características de ambos enfoques.

Una vez que se han determinado las variables relevantes y la técnica de aprendizaje automático adecuada, el objetivo en el problema de clasificación binaria es maximizar su rendimiento mediante el ajuste de los hiperparámetros correspondientes y la aplicación de las índices de desempeño apropiadas para el problema en cuestión.

Por último, se mencionan dos propuestas que podrían ser implementadas en el experimento Belle II con el objetivo de ser consideradas por la colaboración para una posible incorporación en los análisis de decaimientos futuros.

Si bien el aprendizaje automático ya ha demostrado su utilidad en la física de altas energías, con mejoras significativas en diferentes áreas y experimentos, cada proyecto experimental tiene requisitos específicos para alcanzar sus objetivos. Esta tesis se enfoca en la clasificación del decaimiento del leptón tau en el experimento Belle II. El objetivo es tener una buena clasificación de decaimientos, en particular: el decaimiento $\tau \to \pi^+\mu^-\mu^-\nu_\tau$, el cual aún no ha sido observado. Además, la metodología desarrollada puede ser extendida para considerar otros decaimientos que compartan características similares, es decir, aquellos que no presenten cambios en el número de partículas, sino en su tipo.

Partiendo del conocimiento existente sobre el aprendizaje automático [Russell et al., 2004] y las observaciones realizadas en relación con una problemática en particular, a la que nos referiremos como el conjunto de datos, es necesario delimitar nuestro caso de estudio. Esto debido a la diversidad de conjuntos de datos y el vasto conocimiento en el campo del aprendizaje automático, lo cual permitirá apreciar mejor la relación entre las técnicas desarrolladas en este campo y las características de los conjuntos de datos analizados.

Antes de describir los conocimientos y técnicas de aprendizaje automático que serán empleadas en el presente trabajo, es importante comprender el tipo de problema que se desea abordar. Esto permitirá seleccionar las herramientas adecuadas y, si es necesario, adaptarlas para obtener una solución efectiva para el problema en cuestión.

Iniciando con la descripción del conjunto de observaciones o datos que se pretende estudiar, denotado como $D = X_1, X_2, ..., X_m$, donde m representa el número de observaciones o elementos en el conjunto. Se requiere que cada elemento $X_i \in D$ tenga las mismas características, por ejemplo, $X_i \in \mathbb{R}$, $X_i \in \mathbb{R}^2$, o $X_i \in \mathbb{R}^{2\times 7}$, entre otras posibilidades.

Además, se destaca la relevancia del orden interno de cada X_i, excepto en el caso de $X_i \in \mathbb{R}$. Por ejemplo, si $X_i \in \mathbb{R}^2 = (x_1, x_2) \sim (x_2, x_1)$ o $(x_1, x_2) \nsim (x_2, x_1)$, lo que indica si el intercambio de x_1 por x_2 en X_i mantiene su equivalencia a X_i'. En el primer caso, se podrían encontrar conjuntos de datos como formularios de clientes de una empresa o listas de características de un objeto, donde el orden de la información no importa siempre que se respete la información solicitada. En el segundo caso, se encuentran filas ordenadas, representaciones visuales, entre otros casos, donde la posición es relevante, ya que un cambio en la estructura interna de X_i implica una representación diferente. Por ejemplo, al cambiar la posición de los píxeles en una imagen, esta deja de ser la misma imagen.

Es importante mencionar que en este trabajo se enmarca en el primer caso, es decir, podrá aplicarse sobre cualquier conjunto de secuencias finitas, manera más precisa denominas tuplas. Luego, todas las tuplas necesitan ser de la misma longitud, digamos n. Entonces se les denomina n-tuplas. Sin embargo, utilizaremos como referencia conjuntos de datos previamente medidos o generados en estudios anteriores, con el fin de poner a prueba la metodología desarrollada.

Con las características de los conjuntos de datos y los casos particulares establecidos, se determina que el objetivo es extraer la mayor cantidad de información posible. Para ello, utilizaremos el conocimiento del aprendizaje automático, que es una rama de la inteligencia artificial con diversos enfoques. En este trabajo, se opta por el enfoque propuesto por Bellman en 1978 [Russell et al., 2004], que busca la automatización de actividades que se vinculan con los procesos del pensamiento humano, como la toma de decisiones y la resolución de problemas. Debido a que en la actualidad el proceso de estudiar los diferentes decaimientos dentro del experimento Belle II en su gran mayoría se realiza por expertos en la materia

utilizando diversas herramientas estadísticas, lo cual tiene sentido ya que son herramientas que han sido utilizadas por décadas y son parte de las bases de la física moderna. Pero dentro del estudio de los diversos decaimientos existen similitudes que motivan la implementación de técnicas de aprendizaje automático, en particular dentro del proceso de reconstrucción. Esta área de oportunidad se suma a las demás áreas donde existe presencia de inteligencia artificial, mencionadas brevemente en el Capítulo 2. Sin embargo, esto no implica que la presencia de expertos en la materia dejará de existir, todo lo contrario, podrá existir la necesidad de incluir expertos que puedan realizar la conexión entre el aprendizaje máquina y la física de partículas.

Para comenzar con la automatización del proceso de reconstrucción es necesario comenzar con la base, es decir, el conjunto de datos. Para ello, se amplían las características del conjunto $D = X_1, X_2, ..., X_m$ al incluir una componente adicional, denotada como y_i. Cada elemento X_i tendrá la forma $(x_1, x_2, ..., x_n, y_i)$, donde el orden de x_j no es relevante y puede intercambiarse sin cambiar las propiedades de X_i. Sin embargo, la componente y_i puede pertenecer a $\mathbb{R}$ o a cualquier subconjunto de $\mathbb{R}$. Si $y_i \in \mathbb{R}$, se trata de un problema de regresión, mientras que si $y_i \in 0, 1$ u otro conjunto binario, se considera un problema de clasificación (multiclase cuando $y_i \in 0, 1, ..., K$), donde los valores 0 y 1 representan clases independientes. La presencia de la componente y_i en nuestro conjunto de datos nos sitúa en la categoría de problemas supervisados, ya que indica la respuesta esperada de la técnica de aprendizaje automático.

De esta manera es posible determinar si la solución a la problemática ofrecida por la automatización fue correcta o no. En caso de no ser correcta se pueden tomar algunas medidas sobre la forma de solucionar el problema, por ejemplo, revisar más técnicas de aprendizaje automático, cambiar los hiperparámetros de dichas técnicas, ampliar de ser posible el conjunto de datos, entre otras.

Por otro lado, cuando no se conoce la respuesta que debe proporcionar la automatización de la solución, es decir, no contamos con un conocimiento previo y, por lo tanto, no existe la componente y_i se consideran problemas no supervisados. Estos problemas como sus posibles

casos de estudio están fuera del alcance de este trabajo.

En el contexto de problemas supervisados, las componentes o variables x_j de cada instancia X_i son conocidas como variables independientes o predictoras. Estas variables se utilizan como entrada o insumo para las técnicas de aprendizaje automático. La variable y_i es la variable dependiente o objetivo, que representa la respuesta que se espera obtener de la técnica de aprendizaje automático.

En el caso de estudio, se hará énfasis en las técnicas de aprendizaje automático que sean útiles para resolver el problema supervisado. Sin embargo, dado que hay una amplia variedad de técnicas disponibles dado que pueden existir variaciones de una misma técnica, se limitará a utilizar cuatro de ellas con diferentes niveles de complejidad, de tal manera que se realice énfasis en la metodología para la automatización de la reconstrucción y clasificación del presente decaimientos, así como futuros casos de estudio. Además, esto permitirá observar las limitaciones y ventajas de cada una. Las técnicas seleccionadas son: regresión logística, árboles de decisión, máquinas de vectores de soporte y perceptrón multicapa. Estas técnicas se presentarán en el Capítulo 3.

Es importante destacar que la diversidad de técnicas de aprendizaje automático se debe a que no todas tienen el mismo rendimiento al interpretar un mismo conjunto de datos o entorno. En otras palabras, es necesario ajustar la técnica de aprendizaje automático a un entorno específico para extraer la mayor cantidad de información posible.

Al examinar más detenidamente el conjunto de datos D, se pueden observar dos características principales. En primer lugar, la cardinalidad del conjunto de datos, es decir, el número de elementos o instancias, que se denota como $Card(D) = |D| = m$. En segundo lugar, el número de componentes o variables independientes x_j en cada instancia X_i.

Es cierto que al observar con mayor detalle el conjunto de datos, podemos controlar el funcionamiento y rendimiento de las técnicas de aprendizaje automático. Las características del conjunto de datos, tanto la cardinalidad (número de instancias) como el número de variables (independientes) x_j en cada instancia X_i, tienen un impacto significativo en el rendimiento de

las técnicas de aprendizaje aplicadas, no solo en el conjunto D, sino en cualquier conjunto de datos.

En el caso de la cardinalidad del conjunto de datos, pueden surgir diversas problemáticas. Por ejemplo, si el número de instancias es escaso, es posible que no sean suficientes para que la técnica de aprendizaje automático interprete correctamente el entorno y genere resultados precisos. Asimismo, si las instancias no son representativas de la totalidad del entorno y se concentran en sectores específicos, es probable que la técnica no pueda generalizar adecuadamente y los resultados sean sesgados. Además, en problemas de clasificación, puede haber desbalance de clases, lo que significa que una o varias clases pueden estar subrepresentadas en comparación con otras, lo cual puede afectar la capacidad de la técnica para reconocer y clasificar correctamente las clases minoritarias.

En cuanto al número de variables independientes x_j en cada instancia, también pueden surgir desafíos. Si las instancias no contienen las características (variables) suficientes para describir un elemento complejo, como puede ser el caso de describir a un cliente de un banco solo con el nombre, es probable que la técnica no pueda obtener suficiente información para tomar decisiones precisas y, por lo tanto, las respuestas pueden ser erróneas, este problema es llamado subajuste. Por otro lado, si las instancias contienen una gran cantidad de características (variables) para describir un elemento sencillo, la técnica puede enfrentar dificultades para identificar qué características son relevantes provocando problemas de sobreajuste, lo cual dificultará la generalización al considerar nuevos datos.

En este sentido, es de gran interés analizar la relevancia de cada variable x_i en relación con la técnica de aprendizaje utilizada y la respuesta obtenida. Al examinar la importancia de cada variable, podemos identificar cuáles son las más significativas para abordar el problema en cuestión y comprender cómo influyen en las respuestas proporcionadas por las técnicas de aprendizaje automático. Este análisis nos brinda información valiosa sobre las características fundamentales del entorno que estamos estudiando, permitiéndonos tomar decisiones fundamentadas en relación con las variables relevantes para nuestros análisis y

predicciones.

Durante el caso de estudio, que se centra en un problema específico de física de partículas donde la relevancia de cada variable es crucial, podemos explorar con mayor detalle la influencia de cada una de ellas en las técnicas de aprendizaje automático y en las respuestas obtenidas. Esto proporcionará información valiosa para mejorar la comprensión del entorno, optimizar el análisis y realizar predicciones más precisas.

Este problema, tiene contexto en el experimento Belle II, dentro del laboratorio de la Organización de Investigación de Aceleradores de Alta Energía (KEK, por sus siglas en japonés) [Sagawa, 1996]. Este experimento será descrito con más detalle en el Capítulo 4.

El experimento Belle II proporciona un conjunto de datos que se ajusta a nuestras necesidades de estudio, permitiéndonos aplicar técnicas de aprendizaje automático para analizar y comprender mejor el entorno de las partículas detectadas. El conjunto de datos D en este contexto específico cumple con las características que hemos discutido, como el número de instancias y la relevancia de las variables independientes para describir el entorno. Esto brinda una oportunidad única para explorar y aplicar diversas técnicas de aprendizaje automático, con el objetivo de obtener respuestas más precisas y profundizar sobre el conocimiento del mundo de las partículas subatómicas.

Es cierto que este tipo de enfoques transversales se están volviendo cada vez más frecuentes en diferentes áreas, no solo en física, sino también en medicina, finanzas, ciencias sociales y otras disciplinas. Por ejemplo, en el campo de la medicina, se han desarrollado modelos que pueden predecir el riesgo de enfermedades crónicas como la diabetes o el asma, utilizando datos clínicos y registros médicos. En el ámbito financiero, se utilizan técnicas de aprendizaje automático para detectar fraudes en transacciones de tarjetas de crédito al analizar patrones inusuales de gasto. Además, en diversas áreas se utilizan modelos de clasificación basados en datos, como la clasificación de flores en la botánica.

En todos estos ejemplos, el factor común es la presencia de datos como insumos para las técnicas de inteligencia artificial y ciencias de la información. Estos conjuntos de datos pueden

provenir de diversas fuentes, como registros médicos, transacciones financieras o imágenes recopiladas en el campo. Cada uno de estos conjuntos de datos representa un entorno particular y, por lo tanto, requiere un enfoque específico.

Es importante destacar que la aplicación de técnicas de aprendizaje automático en estos contextos transversales ha demostrado ser prometedora y ha generado avances significativos en diversas áreas. Sin embargo, también es fundamental tener en cuenta las particularidades de cada conjunto de datos y considerar cuidadosamente los desafíos y limitaciones asociados. El análisis detallado de cada conjunto de datos y la comprensión de su entorno específico son aspectos clave para lograr resultados precisos y relevantes en la aplicación de técnicas de inteligencia artificial y aunque el número de variables por instancia no es extremadamente alto, las técnicas de aprendizaje automático requieren más recursos computacionales a medida que aumenta la información.

En este caso de estudio, se tiene cierto control sobre el número de instancias por clase, lo que permite abordar el problema de subajuste en cierta medida. Sin embargo, no se conoce la complejidad de las instancias ni sabe si todas las variables son necesarias para describirlas, lo que dificulta el manejo del sobreajuste.

Por lo tanto, se necesita desarrollar una metodología eficiente que cumpla con los requisitos tanto de la física de partículas como del aprendizaje automático. Esto implica describir el problema físico, aplicar diversas técnicas de aprendizaje automático y, finalmente, interpretar los resultados en el contexto original del problema.

La estructura de este trabajo se distribuye de la siguiente manera: en el Capítulo 2, se describe el problema original y los objetivos a alcanzar. En el Capítulo 3, se presenta el marco teórico basado en las ciencias de la información, que incluye la selección de variables y las técnicas de aprendizaje automático que utilizaremos, en el Capítulo 4, se describe brevemente el arreglo experimental de nuestro problema. Con el marco teórico y nuestro caso de estudio establecidos, se presenta la metodología desarrollada durante el presente trabajo en el Capítulo 5. En el Capítulo 6, se presentan los resultados de los métodos de selección de variables y la

comparación de las eficiencias de los clasificadores construidos. En el Capítulo 7, se discuten los resultados obtenidos, se analizan los objetivos alcanzados se presentan las conclusiones, resumiendo los hallazgos principales y discutiendo su relevancia.

Capítulo 2

Antecedentes históricos

El problema abordado en esta tesis se sitúa en el contexto de la física de partículas, específicamente en el experimento Belle II, llevado a cabo en el Laboratorio de Investigación de Aceleradores de Alta Energía (KEK, por sus siglas en japonés) localizado Tsukuba, Japón. El objetivo principal de este experimento es estudiar las propiedades de las partículas subatómicas y comprender mejor las interacciones fundamentales que rigen el universo [Kou et al., 2018, et al. (Particle Data Group), 2014].

En el experimento Belle II, se generan grandes cantidades de datos a partir de colisiones de partículas subatómicas. Estos datos contienen información sobre las características y propiedades de las partículas producidas en las colisiones. El desafío radica en analizar y extraer conocimiento significativo de estos datos para entender mejor los fenómenos físicos subyacentes.

En este contexto, se recurre a las herramientas y técnicas de aprendizaje automático (*machine learning*) para resolver el problema, ya que el aprendizaje automático ofrece enfoques y algoritmos que permiten analizar grandes volúmenes de datos y descubrir patrones, relaciones y regularidades ocultas. Esto difiere del tratamiento tradicional que hace particular énfasis en variables que tienen mayor relevancia dentro de la propia física de partículas, realizando diversos análisis estadísticos para rangos de valores establecidos, véase por ejemplo [Abudinén

and Aggarwal, 2023, Adachi and Ahlburg, 2020, Abudinén and Adachi, 2021].

Es importante destacar que la física de partículas es la rama de la física que estudia los componentes fundamentales o elementales de la materia, así como las interacciones entre ellos. Los resultados de esta área, se pueden pueden resumir de la siguiente manera: la materia está compuesta por dos grupos principales de partículas, **fermiones**, que a su vez se dividen en **quarks** y **leptones**, los cuales son los bloques fundamentales con los que se construye toda la materia y los **bosones**, que son los que dan un orden a los bloques de materia y que permiten formar bloques más complejos [Quintero, 2014]. Un representación gráfica se presenta en la Figura 2.1, en la parte izquierda se encuentran los bloques de materia, los fermiones. Sobre la parte derecha se presentan los mediadores de los bloques de materia, los bosones.

Figura 2.1. Modelo Estándar de Partículas [Commons, 2023].

En la Figura 2.1 se presenta una interpretación del **Modelo Estándar de Partículas** que representa el resultado de diversos experimentos a través del tiempo. Algunos de ellos

realizados en los llamados *aceleradores de partículas* o *colisionadores de partículas*, que como su nombre indica acelera partículas para hacerlas colisionar contra un objetivo fijo o en movimiento, esto con el fin de estudiar los procesos posteriores a la colisión.

A lo largo de los años, la física de partículas ha logrado describir componentes y fuerzas fundamentales de la materia y el Modelo Estándar de Partículas es una de las teorías más comprobadas en este campo. Sin embargo, este modelo no es absoluto y puede haber procesos físicos que no pueden ser explicados por él, lo que podría requerir modificaciones o extensiones. Uno de estos casos es la posibilidad de que los neutrinos tengan una masa diferente de cero, lo cual implicaría la necesidad de extender el Modelo Estándar [López Castro and Quintero, 2013, Quintero, 2014].

En este sentido, el presente trabajo de investigación busca aportar una metodología para seguir poniendo a a prueba el Modelo Estándar, buscando un proceso físico ($\tau^+ \to \pi^-\mu^+\mu^+\bar{\nu}_\tau$) que no ha sido medido de manera experimental pero se estiman las fracciones de decaimiento del orden de 10^-7 [Quintero, 2014], es decir, de aproximadamente 10 millones de pares de taus producidos solo uno par corresponde al decaimiento de interés. Todo esto es posible si los neutrinos tienen masa diferente de cero [Quintero, 2014, López Castro and Quintero, 2013, Mann and Primakoff, 1977, Avignone et al., 2008]. Este proceso físico o decaimiento será buscado en el experimento Belle II [Kou et al., 2018, Sagawa, 1996], el cual será descrito con más detalle en el Capítulo 4.

A pesar de que nuestro problema surge desde el punto de vista de la física de partículas, su solución será buscada dentro del contexto de las ciencias de la información. Existe una conexión entre ambas desde hace varias décadas. Por ejemplo, en la Organización Europea para la Investigación Nuclear [Brüning et al., 2004], se llevan a cabo experimentos con el Gran Colisionador de Hadrones (LHC, por sus siglas en ingles) [LHC,], que requiere de una gran infraestructura informática para almacenar y analizar los datos generados por los diferentes experimentos. En estos experimentos, se han aplicado técnicas de ciencias de datos [de Oliveira et al., 2016, Haake and Loizides, 2019, Cogan et al., 2015, Yu, 2019] para mejorar el rendimiento

de los estudios de física, acelerar los tiempos de cómputo y mejorar la calidad de los datos, entre otros. En el experimento Belle II, también ha realizado estudios con inteligencia artificial, como la reconstrucción de trazas e información de decaimientos de mesones B [Keck et al., 2019].

En resumen, este trabajo de investigación busca aplicar técnicas de aprendizaje automático para analizar los datos generados en el experimento Belle II y extraer información relevante sobre los decaimientos de partículas. Esto contribuirá al avance de la física de partículas y del campo del aprendizaje automático, al tiempo que ayudará a comprender mejor los fenómenos físicos estudiados en el experimento.

Capítulo 3

Marco Teórico

Este capítulo se divide en tres partes centrales. En la primera parte se consideran herramientas para realizar un preprocesamiento del conjunto de datos D. Como ya se ha mencionado, la elección de la técnica de aprendizaje automático con el mejor rendimiento depende en gran medida de las características del conjunto de entrada. En este trabajo, nos enfocamos en seleccionar las variables independientes de los elementos X del conjunto D que maximicen la eficiencia de los clasificadores que construiremos. Estas herramientas de selección se describen en la Sección 3.1.

En la segunda parte, se exploran diversas técnicas de aprendizaje automático que nos permitirán construir diferentes clasificadores. Siendo regresión logística (LR), árboles de decisión (DT), máquinas de vectores soporte (SVM) y perceptrón multicapa (MLP), las técnicas utilizadas durante el presente trabajo y que serán descritas en la Sección 3.2.

Finalmente, en la Sección 3.3 se presentan los métodos y métricas de evaluación para los clasificadores construidos. Estas métricas permiten realizar una comparación entre los clasificadores y seleccionar el más adecuado para el problema establecido. En esta sección se discutirán y aplicarán métricas comunes como la precisión, la exhaustividad, la puntuación F1 y el área bajo la curva ROC. Estas métricas proporcionarán una evaluación objetiva y cuantitativa del desempeño de los clasificadores, lo que facilitará la toma de decisiones

informadas sobre cuál clasificador es el más apropiado para resolver el problema planteado.

Esta manera de abordar la descripción de las herramientas a utilizar se debe a la metodología general que utilizan las técnicas de aprendizaje automático. En la Figura 3.1, se muestra metodología simplificada para la construcción de un clasificador (C) a partir de un conjunto de datos y una técnica de aprendizaje automático (M). Donde el conjunto de datos consta de m número de instancias y cada instancia esta representada por n número de variables independientes. La técnica a utilizar requiere parámetros que necesitan ser definidos por el usuario, los cuales son llamados hiperparámetros (H) y parámetros (P) que tienen que ser ajustados utilizando el conjunto de datos.

En esta metodología, los hiperparámetros son configuraciones que se establecen antes del proceso de entrenamiento y afectan el comportamiento del algoritmo de aprendizaje. Por otro lado, los parámetros son valores internos del modelo que se ajustan durante el entrenamiento para optimizar el rendimiento del clasificador.

Es importante destacar que la elección adecuada de los hiperparámetros y la búsqueda de los mejores valores para los parámetros son aspectos críticos para obtener un "buen" clasificador.

Además, antes de encontrar los parámetros P de la técnica, generalmente se divide el conjunto de datos original en dos subconjuntos: el conjunto de entrenamiento (E) y el conjunto de prueba (T). El conjunto E se utiliza para encontrar los parámetros P de la técnica M y construir así el clasificador C. Una vez que el clasificador ha sido construido, se evalúa su rendimiento utilizando el conjunto de prueba T, que es un conjunto independiente que no se utilizó en el proceso de entrenamiento.

Esta es una descripción simplificada de la construcción de un clasificador, pero nos permite observar que existen diferentes opciones al momento de entrenar una técnica de aprendizaje y construir el clasificador. Primero, es posible construir una gran variedad de clasificadores partiendo de una misma técnica y conjunto de datos. Por ejemplo, seleccionando un número distinto de instancias en los conjuntos de entrenamiento y prueba, $m1$ y $m2$ respectivamente.

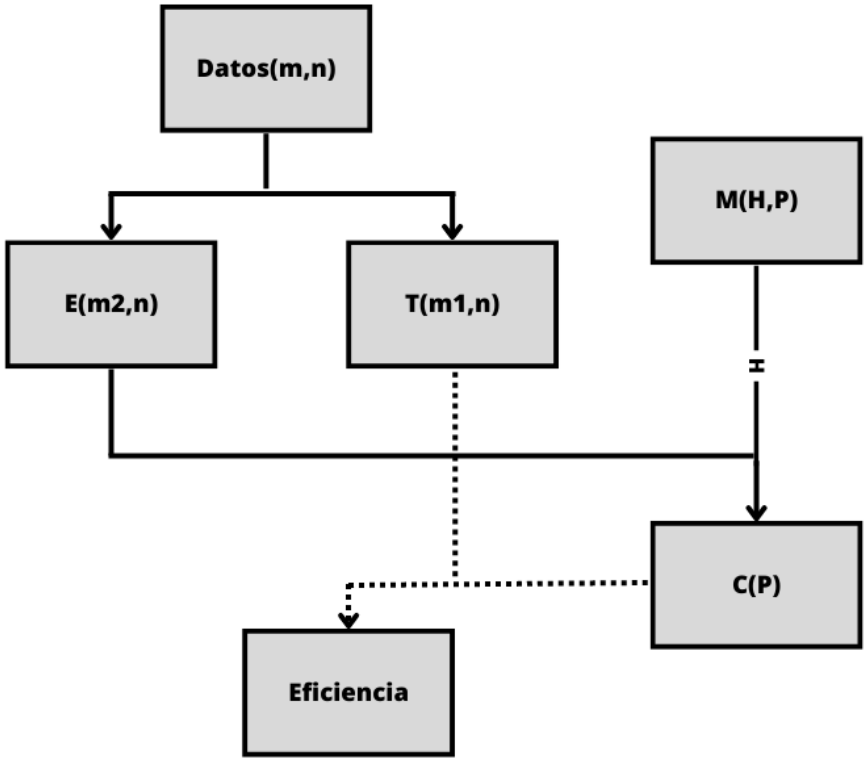

Figura 3.1. Diagrama general del uso de técnicas de aprendizaje automático.

Donde no es necesario que $m1 \cap m2 = m$. Segundo, manteniendo los valores de $m1$ y $m2$ pero cambiando las instancias que los componen. Tercero, cambiando el valor de los hiperparámetros a utilizar puede dar lugar a la construcción de una cantidad infinita de clasificadores. Esto puede ocurrir cuando dichos hiperparámetros son números reales. De manera general se puede establecer que existe un número infinito de clasificadores a construir.

Por lo cual, a pesar de que nuestro problema se centra en encontrar al clasificador con la mayor eficiencia, será imposible evaluar cada uno de los clasificadores para un conjunto de datos. Por lo tanto, se ha optado por limitar el número de clasificadores a construir fijando los valores de los hiperparámetros y el número de instancias en los conjuntos de entrenamiento y prueba. Esto permite enfocar el estudio en las variables que mejor describen a las instancias (Sección 3.1) y que sean relevantes para la eficiencia de los diferentes clasificadores (Sección 3.2).

Esto es un gran desafío, ya que requiere identificar la naturaleza inherente de los datos y

las limitaciones de las técnicas de aprendizaje y que se dificulta cuando se presenta una alta dimensionalidad, ya que provoca una dispersión de los datos y esto dificulta la identificación de patrones significativos o la separación clara entre clases y son aspectos fundamentales para las técnicas de aprendizaje automático, tal como se muestra en la Sección 3.2.

Por último, es importante destacar que existen otros requisitos antes de construir cualquier clasificador, como la ausencia de registros vacíos o indefinidos, ya que estos no pueden ser interpretados por los algoritmos de entrenamiento. Estos problemas suelen surgir durante la adquisición o captura de datos, pero por lo general, pueden resolverse mediante un preprocesamiento sencillo. Algunas estrategias comunes para abordar estos problemas incluyen la eliminación de instancias con registros vacíos o indefinidos, así como la generación de datos aleatorios para completar los registros vacíos, dependiendo de la naturaleza del problema y los datos en cuestión.

3.1. Selección de variables

Considerando que nuestro conjunto de variables supera las 150 variables por instancia, nos encontramos con el desafío de la alta dimensionalidad respecto a las técnicas de aprendizaje, es decir, la variación de la eficiencia de los clasificadores al aumentar el número variables usadas para su construcción, que puede resultar en un sobreajuste de los parámetros de los clasificadores. Este número de variables se debe en parte a la incertidumbre sobre la independencia de las variables independientes con respecto a la variable objetivo y. Lo que significa que utilizar todas las variables independientes para construir un clasificador no garantiza la mejor interpretación del conjunto de datos. Es posible que algunas variables no sean relevantes para nuestro problema, donde dicha relevancia la estableceremos en el Capítulo 5. En general, una variable se considera relevante cuando aporta información útil o significativa para el análisis, la modelización o la comprensión del fenómeno que se está estudiando. Por lo tanto, es importante realizar una selección adecuada de variables o aplicar

técnicas de reducción de dimensionalidad para mejorar la interpretación y el rendimiento del clasificador.

Esto representa un desafío de complejidad exponencial, ya que implica considerar todos los subconjuntos posibles de variables que pueden generarse a partir del conjunto total de variables, es decir, su conjunto potencia $P(n)$, donde n es el conjunto de características o variables que describe cada registro o evento, $X = X(x_1, x_2, ..., x_n)$ (omitiendo la respuesta esperada y). Donde la cardinalidad del conjunto potencia, $|P(n)|$, es de $2^{|n|}$ y ya que para nuestro caso de estudio $n > 150$, nos indica que sería necesario analizar más de 2^{150} conjuntos de variables, lo cual no es factible de considerar debido a las limitaciones de cómputo, pero es posible utilizar algunos métodos de selección de variables que permiten determinar un "buen" conjunto de variables sin la necesidad de analizar la totalidad de los 2^{150} conjuntos. Estos métodos de selección pueden clasificarse dentro de las siguientes categorías: *filtrado*, *envuelto*, *embebido* e *híbridos*. Cada categoría tiene sus propias ventajas y desventajas, y la elección del método adecuado dependerá del problema específico y las características de los datos. A continuación se describen de manera general cada una de las categorías mencionadas.

3.1.1. Métodos de filtrado

Los métodos de filtrado se caracterizan por ser independientes del clasificador que se pretenda entrenar, ya que realizan la selección del conjunto de variables antes del entrenamiento y no se modifican posteriormente. Este tipo métodos suele seleccionar las variables mejor posicionadas de acuerdo a: el cálculo la media, la varianza, la desviación estándar u otra medida estadística, también calculando otras características más complejas como la separación [Tegenfeldt, 2007] y el valor de la información [Zeng, 2013]. Cabe mencionar que en el caso de las medidas estadísticas que no suelen considerar la información de si $X \in A$ o $X \in B$, son llamados métodos no supervisado. Siendo el valor de la información o separación métodos de filtrado supervisado ya que si toman en cuenta lo anterior.

Así, una de las ventajas de este método de selección de variables es que solo se necesita

realizar una vez, posteriormente cualquier clasificador es entrenado con dicho conjunto de variables. Sin embargo, una desventaja es que el usuario es responsable de establecer un valor límite ya sea para la media, varianza o cualquier otra medida que se desee utilizar para la selección, lo que puede llevar a seleccionar un número insuficiente de variables o, por el contrario, incluir variables que no aporten información útil para la construcción del clasificador.

A continuación, se describen brevemente algunas opciones de ranqueo de variables como lo son la varianza, la separación y el valor de la información.

Varianza

Como se mencionó en la Sección 3.1.1, la selección de variables puede ser realizada calculando la varianza [Tegenfeldt, 2007] de cada una de ellas, donde la varianza se define como

$$\sigma_j^2 = \frac{\sum_{i=1}^{m} \omega_i (x_j(i) - \mu_j)^2}{\sum_{i=1}^{m} \omega_i} \tag{3.1}$$

donde m es la cardinalidad del conjunto D, $x_j(i)$ es la componente (variable) j del elemento i por lo que la suma se realiza sobre el vector $\bar{X_j} = (x_j(1), x_j(2), ..., x_j(m))$, μ_j la media de la componente $\bar{X_j}$ ($\mu_j = \frac{\sum_{i=1}^{m} \omega_i (x_j(i))}{\sum_{i=1}^{m} \omega_i}$) y ω_i el peso correspondiente a cada instancia. En el caso de que todas las instancias tengan el mismo peso se tiene que

$$\sigma_j^2 = \frac{\sum_{i=1}^{m} (x_j(i) - \mu_j)^2}{m}. \tag{3.2}$$

Una vez calculada la varianza para cada componente (σ_j^2) se ordenan de manera ascendente y se establece un límite para el valor de la varianza para poder eliminar o seleccionar variables.

Separación

A diferencia del ranqueo por varianza, que no tiene en cuenta la pertenencia de la instancia i a los conjuntos A o B, el cálculo de la separación [Tegenfeldt, 2007] proporciona una medida de separación entre las distribuciones de la variable j teniendo en cuenta los conjuntos A y B.

La medida de separación evalúa la capacidad de una variable para distinguir entre las clases A y B. Cuanto mayor sea la separación, más información aporta la variable para la clasificación. La separación puede expresarse de la siguiente manera

$$\langle S^2 \rangle = \frac{1}{2} \int \frac{(\hat{x_j}_A(x_j) - \hat{x_j}_B(x_j))^2}{\hat{x_j}_A(x_j) + \hat{x_j}_B(x_j)} \tag{3.3}$$

donde $\hat{x_j}_A(x_j)$ y $\hat{x_j}_B(x_j)$ son las distribuciones de la variable $\bar{X}_j$ de los conjuntos A y B, respectivamente. En la ecuación anterior el cálculo de la separación se presenta de forma integral pero puede ser expresada en forma de sumatorio utilizando una cantidad determinada de intervalos (bins) que son divisiones o segmentos que se utilizan para agrupar datos en rangos específicos. Estos intervalos se utilizan para crear histogramas y gráficos de frecuencia, pero dimensiones se dejan a consideración del usuario. A partir de aquí no hay diferencia entre la selección mediante el cálculo de la varianza puesto que también se debe realizar un ordenamiento y establecer un límite para la selección.

Valor de la información

El cálculo del Valor de la Información (IV, por sus siglas en inglés) [Zeng, 2013] al igual que el cálculo de separación, es un método supervisado que toma en cuenta la pertenencia de la instancia j a los conjuntos A o B. Sin embargo, a diferencia del cálculo de separación, el valor de la información trabaja mejor con variables discretas o categóricas, aunque también puede adaptarse para trabajar con variables continuas.

Cuando se trabaja con variables continuas, es necesario convertirlas en variables categóricas mediante la división de su distribución en bines o categorías. La cantidad y límites de los

bines deben ser establecidos por el usuario, lo que implica que se debe decidir de antemano cómo discretizar la variable continua. Una vez realizado este proceso de discretización, se procede a calcular la cantidad llamada *peso de la evidencia* (WOE, por sus siglas en inglés).

El WOE (Ecuación 3.4) es una medida de la fortaleza de la asociación entre la variable independiente y la variable objetivo (clase A o B). Se calcula como el logaritmo natural del cociente entre la probabilidad de pertenecer a la clase A (buena) y la probabilidad de pertenecer a la clase B (mala) dentro de cada bin o categoría de la variable. El WOE puede ser positivo o negativo, dependiendo de si la probabilidad de pertenecer a la clase A es mayor o menor que la probabilidad de pertenecer a la clase B en cada categoría.

$$WOE = \ln \frac{Cantidad\ de\ instancias\ buenas}{Cantidad\ deinstancias\ malas} * 100 \tag{3.4}$$

el valor de la información se calcula sumando los WOE de todas las categorías de la variable. Lo cual se puede expresar de la siguiente manera

$$IV = \sum((Cantidad\ instancias\ buanas - Cantidad\ instancias\ malas) * WOE). \tag{3.5}$$

Cuanto mayor sea el IV, mayor será la capacidad predictiva de la variable para la clasificación.

3.1.2. Métodos de envoltura

Los métodos de envoltura son dependientes de al menos una técnica de aprendizaje, lo que significa que si se modifica la técnica, el proceso de selección de variables debe ajustarse nuevamente. Esto se debe a que los métodos de envoltura utilizan el rendimiento del clasificador como criterio para seleccionar las variables.

Una de las ventajas de los métodos de envoltura es que permiten medir directamente el rendimiento del clasificador con diferentes conjuntos de variables, lo que proporciona una

evaluación más precisa de la calidad de cada conjunto de variables. Esto es posible gracias a que utilizan un proceso iterativo para evaluar diferentes combinaciones de variables y construir sus correspondientes clasificadores.

Sin embargo, esta ventaja también puede ser una desventaja, ya que el proceso de selección puede ser computacionalmente costoso y llevar tiempo, especialmente si el conjunto de datos es grande o si se utilizan técnicas de aprendizaje complejas. Además, el proceso iterativo puede llevar a la construcción de un gran número de clasificadores, lo que puede aumentar el riesgo de sobreajuste al conjunto de entrenamiento.

Es por esto que en la práctica, se busca una estrategia para reducir el espacio de búsqueda y limitar la cantidad de conjuntos de variables a evaluar. Esto podría incluir técnicas de reducción de dimensionalidad, selección de subconjuntos de variables más pequeños y estrategias de selección heurística que intenten encontrar un conjunto de variables de alto rendimiento sin evaluar todas las combinaciones posibles. Por ejemplo, el método de selección paso a paso de *eliminación hacia atrás* y de *selección hacia adelante* [James et al., 2013, Bruce and Bruce, 2017], los cuales se describen en las próximas secciones. Siendo posible construir otros métodos de selección resaltando sus ventajas e implementando nuevas ideas para mejorar su rendimiento.

Por otra parte, la selección de variables con métodos de envoltura puede estar sujeta a la elección de la métrica de rendimiento utilizada para evaluar los clasificadores. Dependiendo de la métrica elegida, diferentes conjuntos de variables pueden considerarse óptimos, lo que puede llevar a decisiones subjetivas o sesgadas en la selección final de variables.

Eliminación hacia atrás

Como ya se ha mencionado, para este método de selección es necesario establecer una técnica de aprendizaje antes de iniciar el proceso. Primero, se construye el clasificador $\mathcal{M}_n$, que utiliza el conjunto completo de n variables. A continuación, creamos $k = n$ clasificadores omitiendo una variable en cada conjunto, es decir, cada clasificador utiliza $n - 1$ variables.

Luego, se selecciona el clasificador con el mejor rendimiento, $\mathcal{M}_{n-1}$, determinado por una función de ajuste elegida por el usuario, y repetimos el paso anterior hasta llegar a un clasificador con solo una variable. Durante este proceso se generan $n(n+1)/2$ clasificadores.

Además, como en cada iteración se selecciona un clasificador $\mathcal{M}_k$, se obtiene un conjunto de n clasificadores finales. Luego, de estos clasificadores, se selecciona el que tenga el mejor rendimiento utilizando la misma función de ajuste elegida por el usuario. El algoritmo de selección por eliminación hacia atrás se muestra en el Algoritmo 1. Para llevar a cabo este algoritmo, es necesario contar con un conjunto de datos D y una técnica de aprendizaje automático $\mathcal{M}$ para construir los clasificadores.

Algorithm 1 Método de Eliminación hacia atrás [James et al., 2013, Bruce and Bruce, 2017].

Require: $D, \mathcal{M}$
Ensure: $\mathcal{M}^*$

1: Si $\mathcal{M}_n$ denota el clasificador *completo*, el cual contiene las n variables predictivas.
2: **for** $k = n-1, ..., 1$: **do**
3: (a) Considerar los k clasificadores que contienen todas menos una de las variables predictivas en $\mathcal{M}_k$ cada uno de ellos con un total de $k-1$ variables.
4: (b) Seleccionar el *mejor* clasificador entre las k opciones y nombrarlo como $\mathcal{M}_{n-1}$. Donde el concepto de *mejor* esta definido por una función de ajuste que puede establecer el usuario.
5: **end for**
6: De los $n+1$ clasificadores construidos $\mathcal{M}_1, .., \mathcal{M}_n$ se selecciona el de mejor rendimiento $\mathcal{M}^*$ usando la función de ajuste.

En la linea 1, se construye un clasificador que considera todas las variables, $\mathcal{M}_n$. En la linea 2, se inicia el proceso de iteración para construir la serie de clasificadores. En la linea 3, se construyen los k clasificadores requeridos. En la linea 4, se selecciona el mejor de la iteración. Una vez terminado el proceso de iteración (linea 5), se realiza una nueva selección en la linea 6.

Selección hacia adelante

A diferencia del método de eliminación hacia atrás, el método de selección hacia adelante comienza con un clasificador que utiliza un conjunto vacío de variables, es decir, un clasificador

constante construido a partir un algoritmo previamente establecido. Luego, construye los n conjuntos posibles de 1 variable cada uno y selecciona aquel con el mejor rendimiento, utilizando una función de ajuste determinada por el usuario. A continuación, repite el proceso anterior para construir los clasificadores que utilizan 2 variables por conjunto y nuevamente selecciona el que presenta el mejor rendimiento. Al finalizar este proceso de iteración, se habrán construido $n(n+1)/2$ clasificadores, de los cuales se tendrá un subconjunto de n clasificadores correspondientes a la mejor opción de cada iteración. Una vez más, se utiliza la función de ajuste para seleccionar el que presenta el mejor rendimiento.

3.1.3. Métodos embebidos

Los métodos embebidos [Quinlan, 1996, Rakotomamonjy, 2003, Genuer et al., 2010] son aquellos que realizan la selección durante la construcción del clasificador, por lo que dependen estrictamente del clasificador a utilizar. En este sentido, sería necesario establecer un método de selección para cada clasificador. Sin embargo, existen técnicas de aprendizaje que son muy utilizadas y que ya incorporan un método de selección por naturaleza. Un ejemplo de ello son los *árboles de decisión* [Quinlan, 1996], ya que en cada iteración intentan particionar el espacio de características de manera que logren una separación homogénea. De esta manera, la variable que logre la mayor homogeneidad en la separación será seleccionada para continuar la construcción del árbol de decisión.

Al igual que en los métodos de filtrado y envoltura, en los métodos embebidos también es necesario una función de ajuste que permita medir el valor de la separación homogénea, además de establecer un límite para el número de particiones a realizar. Algunas de las funciones de ajuste más utilizadas en la literatura son la entropía (ID3) [Quinlan, 1986], la información ganada, el índice Gini, la relación de ganancia (C5.0) [Fürnkranz, ,Quinlan, 1986], la varianza y chi cuadrada [Luchman, 2013].

3.1.4. Métodos híbridos

Un método híbrido se considera aquel que utiliza una combinación de los métodos de selección de variables mencionados anteriormente. Por ejemplo, podría ser la unión de dos métodos de filtrado, la combinación de un método de filtrado y uno envuelto, o cualquier otra posible combinación. De esta manera, es posible aprovechar las mejores características de cada método y crear diferentes enfoques híbridos para la selección de variables.

Un ejemplo de un método híbrido se encuentra en [Hsu et al., 2011, Novakovic et al., 2011], donde utilizan dos métodos envueltos. El primer método sirve como un filtro inicial al emplear un clasificador lineal, y luego aplican el segundo método con un clasificador más complejo, reduciendo así el conjunto de variables de manera más eficiente.

Otro enfoque es el utilizado en [Mandal et al., 2021], donde primero eliminan las variables irrelevantes utilizando el método de filtrado basado en la información mutua entre cada variable predictiva y la variable objetivo, con un valor previamente establecido de la información mutua para la selección. Luego, crean conjuntos de variables para construir árboles neuronales utilizando el método de envoltura, buscando minimizar la redundancia de las variables utilizadas y maximizar el rendimiento de los correspondientes árboles neuronales.

Existen muchos otros ejemplos de métodos híbridos en la literatura, como [Jain and Singh, 2018, Viharos et al., 2021, Mohtashami and Eftekhari, 2018, Austin and Tu, 2004], entre otros. Estos enfoques combinados pueden proporcionar soluciones más eficaces y eficientes para problemas específicos de selección de variables en el aprendizaje automático.

Nuestro método se encuentra dentro de esta categoría al establecer un punto de referencia para la calidad de un clasificador, basándose en el método de eliminación a hacia atrás. Además, se busca determinar el conjunto con el menor número de variables que permita construir un clasificador con estas características, utilizando el método de ordenamiento por valor de la información. Esta metodología será presentada con mayor detalle en el Capitulo 5.

3.2. Técnicas de aprendizaje automático

La segunda parte de la metodología se enfoca en las técnicas de aprendizaje automático y la construcción de los clasificadores que serán utilizados en el presente trabajo. Es importante destacar que, aunque durante el proceso de selección de variables es posible hacer uso de los clasificadores, no es necesario que el clasificador final sea uno de ellos. Por ejemplo, se puede seleccionar las variables utilizando clasificadores de regresión logística y buscar el mejor rendimiento entre diferentes máquinas de vectores soporte. Por lo tanto, estas técnicas de construcción de clasificadores se describen de manera independiente al proceso de selección de variables.

Comenzaremos describiendo de manera extensa el modelo de regresión logística, ya que es un modelo basado en la teoría de probabilidad y se ajusta a la naturaleza de nuestro problema. Luego, se describirán brevemente los modelos de árboles de decisión, máquinas de vectores soporte y perceptrón multicapa. Estas descripciones estarán enfocadas en el problema de clasificación binaria y no en su versión de regresión.

Con estas técnicas y los conjuntos de datos seleccionados, se procede con el entrenamiento para construir los clasificadores necesarios. Por lo tanto, será fundamental tener una medida del rendimiento de cada clasificador, para lo cual se describirán algunas métricas que pueden ser utilizadas. Estas métricas se basan en la "Curva de Característica Operativa del Receptor" (curva ROC) y la "matriz de confusión". La curva ROC nos permitirá seleccionar el clasificador con el mejor rendimiento global, mientras que la matriz de confusión será útil para ajustar las condiciones del clasificador previamente seleccionado, de acuerdo a las necesidades específicas del usuario. Estas necesidades suelen estar relacionadas con las características del problema a resolver.

3.2.1. Modelo de regresión logística

Como se ha mencionado, se comienza describiendo de manera amplia el modelo de Regresión Logística [Cox, 1958] para entender el contexto estadístico del problema.

Este modelo se utiliza para resolver problemas de clasificación binaria, es decir, aquellos en los que existen dos posibles resultados: éxito o fracaso. Se basa en la distribución binomial y se aplica cuando tenemos una serie de ensayos o pruebas donde cada elemento puede clasificarse en una de dos categorías.

Para que un problema pueda ser aproximado con una distribución binomial, es necesario que se cumplan las siguientes condiciones para cada elemento (ensayo) del conjunto de datos:

- En cada ensayo solo deben existir dos resultados posibles, éxito o fracaso.
- La probabilidad de éxito debe ser constante.
- La probabilidad de fracaso también debe ser constante.
- Cada ensayo debe ser independiente.
- Los ensayos son mutuamente excluyentes, es decir, no pueden ocurrir los dos al mismo tiempo.

Nuestro problema de física de partículas cumple con las condiciones necesarias para ser aproximado con una distribución binomial. Cada ensayo, que representa las colisiones electrón-positrón, es independiente y reproducible dentro del experimento. Además, la probabilidad de cada decaimiento en el experimento es constante, teniendo en cuenta los errores de medición.

Podemos considerar el caso de éxito como aquel en el que un elemento $X \in D$ pertenece al conjunto A, y el caso de fracaso cuando pertenece al conjunto B. Es importante destacar que una vez medido experimentalmente el elemento $X \in D$ no puede pertenecer simultáneamente a los conjuntos A y B, es decir, $A \cap B = \emptyset$.

Dadas estas condiciones, el uso del modelo de regresión logística parece ser el enfoque adecuado para abordar nuestro problema.

Regresión logística

Dado un conjunto de datos, $D = \{(x_1, y_1), ..., (x_m, y_m)\}$, compuesto de variables predictivas, $x_i \in \mathbb{R}^n$ y una variable objetivo, $y_i \in \{0, 1\}$, el modelo de regresión logística pretende clasificar la variable objetivo a partir de las variables predictivas. Donde estamos considerando que las variables predictivas pertenecen al conjunto de los números reales ($\mathbb{R}$). Sin embargo, es posible extender la descripción para incluir variables categóricas dentro del conjunto de variables predictivas.

Entonces, la regresión logística [Cox, 1958] estima cada y_i utilizando la distribución de Bernoulli que tiene la siguiente forma

$$P(y = y_i) = p^{y_i}(1 - p)^{(1-y_i)} \tag{3.6}$$

donde la probabilidad de éxito (p) se aproxima mediante la función sigmoide $\sigma(a) = \frac{1}{1+e^{-a}}$ en términos del producto escalar o combinación lineal de los vectores $X_i = (1, x_i)$, es decir, $a = \omega^T X_i$. Por lo tanto, la probabilidad de que y_i pertenezca a la clase 1 está dada por:

$$P(y_i = 1|x_i, \omega) = \sigma(\omega^T x_i) = \frac{1}{1 + e^{-\omega^T x_i}} \tag{3.7}$$

De manera similar, la probabilidad de que y_i pertenezca a la clase 0 (fracaso) es:

$$P(y_i = 0|x_i, \omega) = 1 - P(y_i = 1|x_i, \omega) = \frac{e^{-\omega^T x_i}}{1 + e^{-\omega^T x_i}} \tag{3.8}$$

Siendo $\omega \in \mathbb{R}^{n+1}$ el vector que contiene la información del modelo de regresión logística y que necesita ser estimado. Para ello, se utiliza el método de estimación de máxima verosimilitud [Aldrich, 1997], es decir, encontrar el vector ω_{rl} que maximice la función $P(D|\omega)$. Lo cual se puede escribir como

$$\omega_{rl} = max(P(D|\omega)) \tag{3.9}$$

donde

$$P(D|\omega) = \prod_{i=1}^{m} p(y_i|x_i, \omega) \tag{3.10}$$

y de acuerdo a la distribución de Bernoulli (3.6) se tiene que

$$P(D|\omega) = \prod_{i=1}^{m} p(y_i|x_i, \omega) = \prod_{i=1}^{m} \alpha_i^{y_i}(1-\alpha_i)^{1-y_i}. \tag{3.11}$$

donde $\alpha_i = \sigma(\omega^T X_i)$. Sin embargo, en lugar de maximizar la función de verosimilitud (3.11), en la regresión logística se suele trabajar con su logaritmo para simplificar considerablemente el cálculo y la manipulación de funciones como se observará a continuación. Además, en muchos problemas de optimización, buscar el mínimo es más estable y común que encontrar un máximo, por lo que se trabaja con la función logarítmica negativa de la verosimilitud, que se denota como $\mathcal{L}(\omega)$ y se expresa de la siguiente forma:

$$\mathcal{L}(\omega) = -log(P(D|\omega)) = -\sum_{i=1}^{m} \left(y_i log(\alpha_i) + (1-y_i) log(1-\alpha_i)\right). \tag{3.12}$$

Por otra parte, recordando que el mínimo de una función implica utilizar las derivadas parciales respecto a las variables dicha función (ω_i's), es decir,

$$\frac{\partial \mathcal{L}(w)}{\partial w_j} = -\sum_{i=1}^{m} \left(y_i \frac{\partial log(\alpha_i)}{\partial w_j} + (1-y_i) \frac{\partial log(1-\alpha_i)}{\partial w_j} \right) \tag{3.13}$$

donde las derivadas parciales de $log(\alpha_i)$ y $log(1-\alpha_i)$ se pueden escribir como

$$\frac{\partial log(\alpha_i)}{\partial w_j} = x_{ij}(1-\alpha_i) \tag{3.14}$$

$$\frac{\partial log(1-\alpha_i)}{\partial w_j} = -\alpha_i x_{ij}. \tag{3.15}$$

Ahora, sustituyendo (3.14) y (3.15) en (3.13) obtenemos

$$\frac{\partial \mathcal{L}(w)}{\partial w_j} = -\sum_{i=1}^{m} \left(y_i x_{ij} (1 - \alpha_i) + (1 - y_i)(-\alpha_i x_{ij}) \right), \tag{3.16}$$

ecuación que puede reescribirse como

$$\frac{\partial \mathcal{L}(w)}{\partial w_j} = \sum_{i=1}^{m} \left(\alpha_i - y_i \right) x_{ij}. \tag{3.17}$$

Además, dado que (3.12) puede generalmente es una función de más de una variable es necesario considerar el gradiente, es decir, el conjunto de las derivadas parciales de la función en ese punto

$$\begin{aligned} \nabla \mathcal{L}(\omega) &= \left(\frac{\partial \mathcal{L}(\omega)}{\partial \omega_0}, \frac{\partial \mathcal{L}(\omega}{\partial \omega_1}, ..., \frac{\partial \mathcal{L}(\omega)}{\partial \omega_n} \right) \\ &= \left(\sum_{i=1}^{m} \left(\alpha_i - y_i \right) x_{i0}, \sum_{i=1}^{m} \left(\alpha_i - y_i \right) x_{i1}, ..., \sum_{i=1}^{m} \left(\alpha_i - y_i \right) x_{in} \right) \end{aligned} \tag{3.18}$$

la cual puede ser escrita en forma vectorial como

$$\nabla \mathcal{L}(\omega) = ((\alpha - y) \cdot \bar{x}_0, (\alpha - y) \cdot \bar{x}_1, ..., (\alpha - y) \cdot \bar{x}_n) \tag{3.19}$$

donde $\alpha^T = (\alpha_1, \alpha_2, ..., \alpha_m)$, $y^T = (y_1, y_2, ..., y_m)$ y $\bar{x}_j = (x_{1j}, x_{2j}, ..., x_{mj})$ con $j = 0, 1, ..., n$, además (3.18) debe ser igualada a cero, ya que el mínimo (máximo) de una función ocurren cuando su derivada se anula, por lo que se obtenemos

$$\nabla \mathcal{L}(\omega) = \mathbf{A^T}(\alpha - \mathbf{y}) = \mathbf{0} \tag{3.20}$$

con

$$A = \begin{pmatrix} x_{10} & x_{11} & \cdots & x_{1n} \\ x_{20} & x_{21} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m0} & x_{m1} & \cdots & x_{mn} \end{pmatrix}. \tag{3.21}$$

Con lo que resta encontrar los valores de α_i que determinen la igualdad de la Ecuación 3.20, de tal manera que sea posible modelar la probabilidad de éxito al utilizar la función sigmoide $\alpha = \sigma(a_i) = \frac{1}{1+e^{-a_i}}$, recordando que a_i es el producto escalar o combinación lineal de los vectores de coeficientes $\omega = (\omega_0, \omega_1, ..., \omega_n)$ y las variables predictivas $x_i = (1, x_{i1}, x_{i2}, ..., x_{in})$.

Proceso de optimización

A pesar de la forma vectorial de la Ecuación 3.20, no siempre es posible utilizar las técnicas de álgebra lineal para encontrar una solución única del vector α. Esto se debe a que generalmente dentro de nuestro contexto la matriz $\mathbf{A}$, que representa la relación lineal entre el número instancias del conjunto de D y el número de variables de cada instancia, no necesariamente es cuadrada, lo que implica que no tenemos el mismo número de ensayos que de variables a determinar. Condición necesaria para obtener una solución única. En consecuencia, pueden existir diversas soluciones para el vector ω_{rl}.

Por lo tanto, para encontrar una solución que minimice el problema, se utiliza una aproximación de $\mathcal{L}(\omega)$ (Ecuación 3.12) mediante una expansión en serie de Taylor de orden 2 para decidir cómo actualizar la posición actual en la búsqueda del mínimo (máximo) alrededor de un vector constante s cercano a este mínimo local (o global). Esto nos lleva a la siguiente expresión:

$$\mathcal{L}(\omega) \approx \mathcal{L}(s) + \nabla\mathcal{L}(s)(\omega - s) + \frac{1}{2}(\omega - s)^T H(\mathcal{L}(s))(\omega - s) \tag{3.22}$$

para después utilizar el método de optimización de Newton [Ypma, 1995], que es actualizar la posición actual siguiendo la dirección en la que se espera se reduzca más rápidamente el valor

de la función. Pero antes describiremos la matriz Hessiana ($H(\mathcal{L})$) de la Ecuación 3.12,

$$\frac{\partial \mathcal{L}(w)}{\partial w_j \partial w_k} = \frac{\partial}{\partial w_j}\left[\sum_{i=1}^{m}(\alpha_i - y_i)\, x_{ik}\right] = \sum_{i=1}^{m} x_{ik}\frac{\partial \alpha_i}{\partial w_j} \tag{3.23}$$

que puede ser reescrita como

$$\frac{\partial \mathcal{L}(w)}{\partial w_j \partial w_k} = \sum_{i=1}^{m} x_{ik}x_{ij}\alpha_i(1-\alpha_i) \tag{3.24}$$

permitiendo escribir la matriz Hessiana de la siguiente forma

$$H(\mathcal{L}(s)) = \begin{pmatrix} \sum x_{i0}\alpha_i(1-\alpha_i)x_{i0} & \sum x_{i0}\alpha_i(1-\alpha_i)x_{i1} & \cdots & \sum x_{i0}\alpha_i(1-\alpha_i)x_{id} \\ \sum x_{i1}\alpha_i(1-\alpha_i)x_{i0} & \sum x_{i1}\alpha_i(1-\alpha_i)x_{i1} & \cdots & \sum x_{i1}\alpha_i(1-\alpha_i)x_{id} \\ \vdots & \vdots & \ddots & \vdots \\ \sum x_{id}\alpha_i(1-\alpha_i)x_{i0} & \sum x_{id}\alpha_i(1-\alpha_i)x_{i1} & \cdots & \sum x_{id}\alpha_i(1-\alpha_i)x_{id} \end{pmatrix} \tag{3.25}$$

donde $\sigma_i = \sigma_i(s)$, además es posible escribir su forma vectorial de la siguiente manera

$$H(\mathcal{L}(s)) = \mathbf{A^T B A} \tag{3.26}$$

donde $\mathbf{A}$ corresponde a (3.21) y

$$\mathbf{B} = \begin{pmatrix} \alpha_1(1-\alpha_1) & 0 & \cdots & 0 \\ 0 & \alpha_2(1-\alpha_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_n(1-\alpha_n) \end{pmatrix}. \tag{3.27}$$

Con la matriz Hessiana descrita, aplicamos la idea de optimización de Newton para encontrar el mínimo de la aproximación de $\mathcal{L}(w)$ (Ecuación 3.22). Para ello, se repete el procedimiento de derivar respecto al vector ω e igualar a cero la derivada resultante; con lo

que obtenemos la siguiente expresión:

$$\nabla\mathcal{L}(\omega) \approx \nabla\mathcal{L}(\mathbf{s}) + H(\mathcal{L}(\mathbf{s}))\omega - H(\mathcal{L}(\mathbf{s}))\mathbf{s} = 0 \tag{3.28}$$

lo cual permite despejar ω de la siguiente manera

$$\omega = \mathbf{s} - H^{-1}(\mathcal{L}(\mathbf{s}))\nabla(\mathcal{L}(\mathbf{s})). \tag{3.29}$$

Para escribir el algoritmo de optimización de Newton, es importante recordar que la matriz Hessiana es definida positiva, ya que se está considerando un mínimo local (global). Por lo tanto, la diferencia en la Ecuación 3.29 nos aproxima a encontrar el mínimo deseado. Esto se logra mediante un proceso de iteración en el que se actualiza ω como ω_{t+1} y s como ω_t. La actualización se expresa de la siguiente manera:

$$\omega_{\mathbf{t+1}} = \omega_{\mathbf{t}} - H^{-1}(\mathcal{L}(\omega_t))\nabla\mathcal{L}(\omega_t) \tag{3.30}$$

o escrito en forma matricial

$$\omega_{\mathbf{t+1}} = \omega_{\mathbf{t}} - [(A^T B A)^{-1} A^T(\alpha - \mathbf{y})]_t. \tag{3.31}$$

Es importante mencionar que el resultado previo no es el caso más general, ya que existen situaciones particulares en las cuales no sería posible resolver la Ecuación 3.20, por ejemplo, cuando la inversa de la matriz Hessiana $H^{-1}(\mathcal{L}(\omega_t))$ no existe. En esos casos, sería necesario ampliar el método o buscar otras técnicas para encontrar una solución.

No obstante, para fines prácticos, se considera que la inversa de la matriz Hessiana existe o que es posible establecer otros métodos de optimización que permitan resolver el problema de manera efectiva. En la práctica, existen diversas variantes y algoritmos de optimización que pueden ser empleados para encontrar una solución aproximada en casos más complejos donde la inversa de la matriz Hessiana pueda ser difícil de calcular o cuando no sea definida

positiva.

En este punto, se podría considerar que el método de regresión logística es una opción adecuada para encontrar la solución a nuestro problema. Sin embargo, es importante destacar que este método asume que las variables predictivas son completamente independientes, lo cual no siempre se cumple o puede ser difícil de determinar. Por lo tanto, la Ecuación 3.13 podría no ser válida y requeriría modificaciones en el desarrollo posterior.

Aquí es donde se aprecia la importancia de seleccionar las variables adecuadas para construir un clasificador eficiente, como se mencionó en la Sección 3.1. Esta selección puede realizarse antes o durante el entrenamiento del clasificador, y resulta fundamental para obtener resultados precisos y confiables.

Esto no implica que el modelo de regresión logística sea una mala alternativa o deba ser descartado, ya que no existe una técnica o clasificador que resuelva todos los tipos de problemas de manera óptima. Sin embargo, el modelo ofrece herramientas estadísticas que permiten mitigar el problema de selección de variables, como el uso de pruebas de hipótesis [Lehmann, 1993] para medir el nivel de confianza de las diferentes distribuciones estadísticas utilizadas, así como evaluar la importancia que cada variable tiene dentro del clasificador.

Contraste de hipótesis y valor de p

Una de las ventajas que nos ofrece el modelo de regresión logística es la capacidad para realizar pruebas de hipótesis, las cuales miden la relevancia que cada variable tiene dentro del clasificador construido. Estas pruebas se basan en los coeficientes del vector ω, los cuales dependen de la muestra utilizada para entrenar el clasificador y, por lo tanto, sus valores pueden variar al utilizar diferentes muestras. Las pruebas de hipótesis [Lehmann, 1993] son de gran ayuda para abordar este problema.

En particular, se plantea una hipótesis nula cuando un coeficiente toma el valor 0, lo que implica medir el rendimiento del clasificador en ausencia de la variable correspondiente. Esta hipótesis nula será contrastada con la hipótesis alternativa, que considera el valor del

coeficiente obtenido después del entrenamiento.

El resultado de esta prueba de hipótesis es un valor dentro del rango $[0, 1]$ conocido como *valor de p*. Para interpretar esta información, se establece que valores cercanos a 0 corresponden a la hipótesis nula y valores cercanos a 1 a la hipótesis alternativa. Sin embargo, es responsabilidad del usuario definir el umbral de cercanía para rechazar o aceptar una de las dos hipótesis.

En un proceso de selección de variables, el objetivo es eliminar aquellas variables cuyo valor de p sea cercano a cero, lo que implicaría aceptar la hipótesis nula y, por lo tanto, indicaría que la variable no tiene relevancia en la construcción del clasificador.

Es importante mencionar que, generalmente, se acepta la hipótesis nula cuando el valor de p es menor a 0.05. Sin embargo, este límite puede ser más alto si el número de variables predictoras es extenso. Es necesario adaptar el umbral según el contexto y los requisitos específicos del problema.

3.2.2. Árboles de decisión

Los árboles de decisión [Russell et al., 2004] son algoritmos de aprendizaje que se basan en secuencias de condiciones con el propósito de alcanzar una decisión final. Estas secuencias de condiciones pueden ser representadas gráficamente haciendo una analogía con la estructura de un árbol (invertido), donde cada condición da lugar a un nodo, una serie de condiciones a ramas y las respuestas finales a hojas. Un ejemplo de un árbol de decisiones se presenta en la Figura 3.2, donde podemos observar que el nodo inicial es llamado raíz y cada rama debe terminar en una decisión u hoja.

En el proceso de construcción de un árbol de decisión [Russell et al., 2004], es esencial determinar el orden de los nodos o condiciones y la profundidad del árbol para lograr una clasificación óptima de las clases. El objetivo es encontrar la mejor manera de dividir el conjunto de datos en subconjuntos más pequeños y homogéneos en términos de la variable objetivo (la clase que queremos predecir). Para lograr esto, se utilizan diferentes algoritmos y

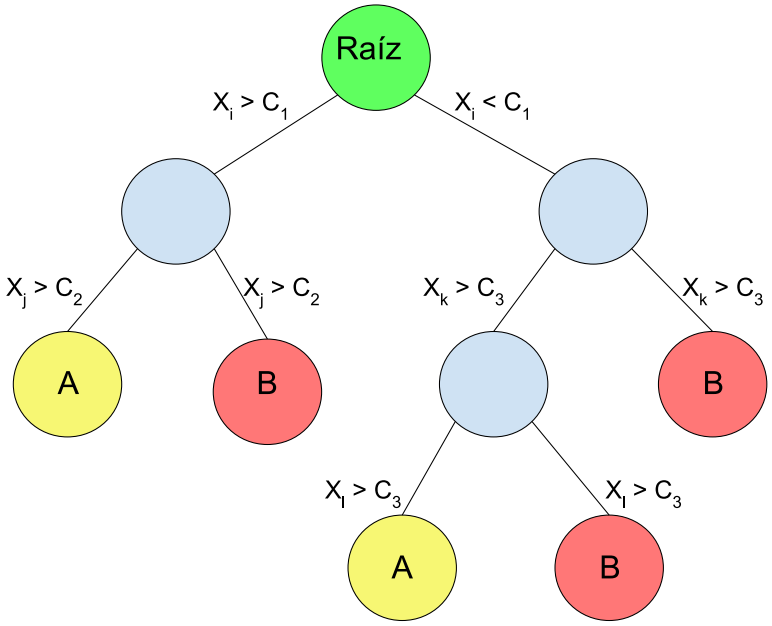

Figura 3.2. Representación gráfica de un árbol de decisión binario, el cual busca determinar si la instancia considerada pertenece a la clase A o B. Esto a partir de una serie de decisiones que pueden tomar forma de árbol.

medidas para evaluar la importancia de cada variable predictiva y decidir qué variable y en qué punto se debe realizar cada división. Siendo algunos de los métodos más utilizados la entropía (ID3) [Quinlan, 1986], el índice Gini [Fürnkranz,], la relación de ganancia (C5.0) [Quinlan, 1986], la varianza y CHAID [Luchman, 2013].

Optimización

La idea de optimización para el árbol de decisión [Russell et al., 2004, Bishop, 2006] es encontrar el árbol más pequeño que pueda discriminar mejor las clases en un conjunto. Optar por la opción trivial de construir un árbol que clasifique cada elemento del conjunto de entrenamiento no será útil para tratar nuevas instancias o datos no vistos previamente. Por lo tanto, es preferible seleccionar de manera descendente las variables o atributos (x_i) que mejor discriminan las clases del conjunto (D), lo que resultará en árboles más pequeños y eficientes.

A continuación, se menciona una manera de seleccionar los mejores atributos .

Partiendo del hecho de que el mejor atributo que divide las instancias en sus clases correspondientes es difícil de encontrar, es necesario establecer criterios para determinar cuándo un atributo es "bastante adecuado" o "inadecuado" y cuánta información se gana con cada decisión tomada. Para ello, es posible comenzar con la información proporcionada por el propio conjunto de datos, independientemente de cualquier variable o atributo, y utilizarla a lo largo de los niveles del árbol [Russell et al., 2004, Bishop, 2006].

Para cuantificar la información ganada después de cada decisión, se calcula el contenido de la información (Ecuación 3.32), que estima la probabilidad de obtener una clasificación correcta, representando la información disponible. La expresión es la siguiente:

$$I\left(\frac{a}{a+b}, \frac{b}{a+b}\right) = -\frac{a}{a+b}\log_2\frac{a}{a+b} - \frac{b}{a+b}\log_2\frac{b}{a+b} \tag{3.32}$$

donde a y b son las instancias de la clase A y B, respectivamente. Al utilizar información de una determinada variable, quedará menos información disponible para la clasificación. La cantidad de información perdida dependerá tanto de la variable seleccionada (x_k) como del valor de la condición empleada para dividir el conjunto de datos (D). Al considerar los diferentes valores de un atributo x_k, es posible dividir el conjunto D en una serie de subconjuntos D_1, D_2, ..., D_μ. Cada uno de estos subconjuntos contendrá la información correspondiente $I(\frac{a_\nu}{a_\nu+b_\nu}, \frac{b_\nu}{a_\nu+b_\nu})$, donde a_ν y b_ν son las instancias de las clases A y B, respectivamente, para el subconjunto ν.

Por lo tanto, es posible calcular el resto de información $R(x_k)$ para construir el árbol. Esta medida representa la cantidad de información que queda después de considerar la variable x_k y se expresa de la siguiente manera:

$$R(x_k) = \sum_{\nu}^{\mu} \frac{a_\nu + b_\nu}{a+b} I\left(\frac{a_\nu}{a_\nu + b_\nu}, \frac{b_\nu}{a_\nu + b_\nu}\right). \tag{3.33}$$

Siendo la diferencia entre la cantidad de información (Ecuación 3.32) disponible y el resto de información (Ecuación 3.33) la ganancia de información (Ecuación 3.34) que ofrece cierto

atributo dada un condición, la cual expresamos como

$$G(x_k) = I\left(\frac{a}{a+b}, \frac{b}{a+b}\right) - R(x_k). \tag{3.34}$$

Esto nos permite seleccionar las variables con mayor ganancia de información y, al mismo tiempo, limitar el tamaño del árbol. Sin embargo, existen otras formas de optimizar el árbol de decisión así como se mencionó en la Sección 3.2.2.

3.2.3. Máquinas de vectores soporte

Las máquinas de vectores soporte [Russell et al., 2004, Bishop, 2006, Cristianini and Ricci, 2008] son modelos de aprendizaje supervisado que buscan encontrar un hiperplano en el hiperespacio definido por el conjunto de entrenamiento $D = \{(x_1, y_1), ..., (x_m, y_m)\}$, de tal manera que sea posible separar las clases del conjunto en cada semi-hiperespacio delimitado por dicho hiperplano. Una representación gráfica de este concepto se muestra en la Figura 3.3, donde una recta puede separar ambas clases, lo cual representa un caso linealmente separable, que rara vez ocurre en la práctica. Por lo tanto, se deben considerar enfoques más complejos para abordar casos no lineales. Sin embargo, al observar esta situación, se pueden apreciar las características principales de las máquinas de vectores soporte.

Entonces, la idea principal de las máquinas de vectores soporte es suponer que existe un separador lineal en el hiperespacio capaz de dividir ambas clases. Por lo tanto, se busca encontrar un hiperplano que maximice la distancia entre los puntos más cercanos de cada clase, lo que se conoce como el margen. Este hiperplano se convierte en la frontera de decisión que separa las clases de manera óptima. En el caso de datos linealmente separables, esto es posible y relativamente sencillo, ya que el hiperplano se puede expresar de la siguiente manera:

$$f(\mathbf{X}) = \sum_{i=1}^{n} w_i x_i + b = \mathbf{W} \cdot \mathbf{X} + b \tag{3.35}$$

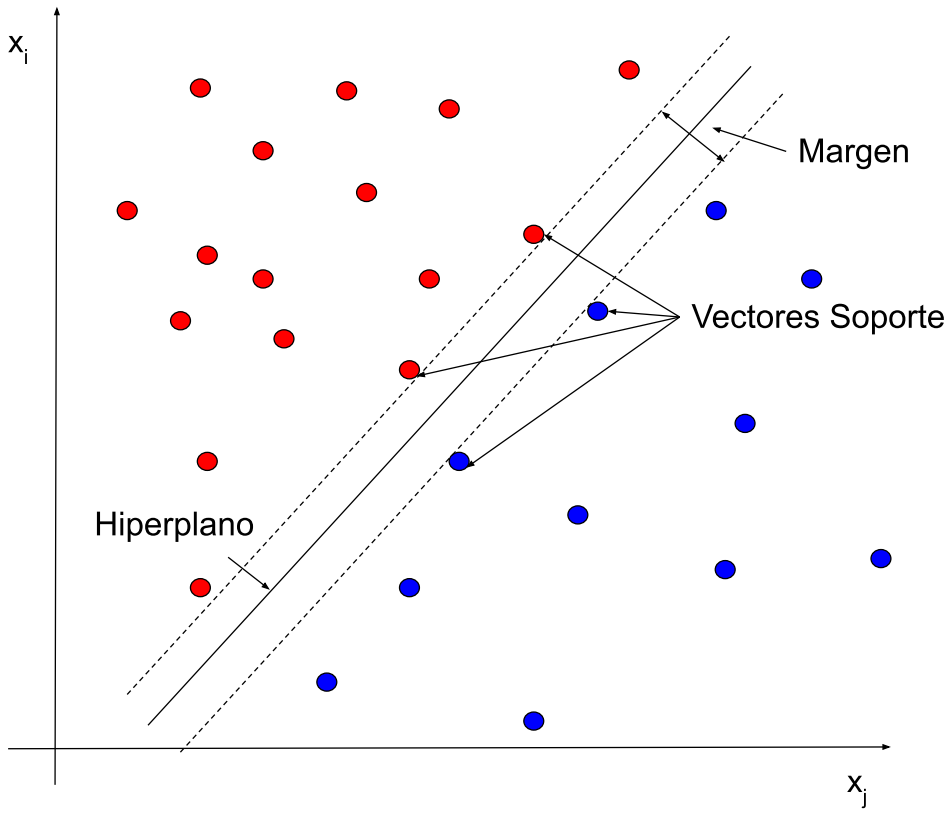

Figura 3.3. Representación gráfica de una máquina de vectores soporte en el espacio de variables x_i y x_j, la cual busca el hiperplano que maximice el margen entre los vectores soporte.

donde se espera que si $f(\mathbf{X}) \geqslant 0$ entonces $\mathbf{X}$ pertenece a la clase positiva y, de lo contrario, a la clase negativa. Definiendo el margen como:

$$\gamma_i = y_i(w_i x_i + b) = y_i(\mathbf{W} \cdot \mathbf{X_i} + b). \tag{3.36}$$

con el objetivo de ser maximizado respecto al hiperplano y los vectores $\mathbf{X_i}$ (Figura 3.3). Esto se convierte en un problema de optimización sobre el vector γ y es llamado "problema principal", ya que considera la optimización del margen sobre el hiperplano $(\mathbf{W}, b)$ y las respectivas clases. Proceso que como en la mayoría de los modelos de aprendizaje supervisado requiere de una función de error o coste y un método de aproximación, los cuales se mencionan

brevemente en la siguiente sección.

Optimización

Para encontrar el plano que maximice la función margen, necesitamos una función de error generalizada. Esto se logra mediante un enfoque estadístico, dando una estimación de la probabilidad de clasificación en términos de la distancia del punto de entrada $\mathbf{X_i}$ al hiperplano ($\mathbf{W}$, b). Esta función de error (para más detalles consultar [Cristianini and Ricci, 2008]) puede escribirse como:

$$L(\mathbf{W}, \mathbf{b}, \alpha) = \frac{1}{2}\mathbf{W} \cdot \mathbf{W} + \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{W} \cdot \mathbf{X_i} + b) - 1]. \tag{3.37}$$

En esta función, se considera la normalización de $\mathbf{W}$ y los multiplicadores de Lagrange $\alpha_i \geqslant 0$. Estos multiplicadores son necesarios ya que también se modifican las condiciones de la función de margen $y_i(\mathbf{W} \cdot \mathbf{X_i} + b) - 1 \geqslant 1$. Este proceso puede considerarse como el caso análogo de pasar del modelo de regresión lineal al modelo de regresión logística para estudiar los problemas de regresión y clasificación, respectivamente. Por lo tanto, el siguiente paso consiste en usar las ecuaciones de Lagrange para minimizar dicha función de error [Cristianini and Ricci, 2008].

$$\frac{\partial L(\mathbf{W}, b, \alpha)}{\partial \mathbf{W}} = \mathbf{0} = \mathbf{W} - \sum_{i=1}^{n} y_i \alpha_i \mathbf{X_i} \tag{3.38}$$

$$\frac{\partial L(\mathbf{W}, b, \alpha)}{\partial \mathbf{b}} = 0 = \sum_{i=1}^{n} y_i \alpha_i \tag{3.39}$$

donde los resultados se sustituyen en la Ecuación 3.37, lo cual se puede reescribir como

$$L(\mathbf{W}, b, \alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j \mathbf{X_i} \cdot \mathbf{X_j} \tag{3.40}$$

Haremos énfasis tanto en la Ecuación 3.40 como en la Ecuación 3.38, ya que implican que

encontrar el plano ($\mathbf{W}$, b) que maximice los margenes es equivalente a encontrar los vectores $\mathbf{X_j^*}$ que definan al vector $\mathbf{W}$

$$\mathbf{W} = \sum_{j=1}^{n} \alpha_j y_j \mathbf{X_j^*} \tag{3.41}$$

estos vectores son los llamados *vectores soporte*, y de ahí proviene el nombre de este modelo de clasificación.

Es importante mencionar que esta descripción es válida únicamente para datos que son linealmente separables en el hiperplano. Sin embargo, en muchos casos, los datos no son linealmente separables, por lo que es necesario construir máquinas más complejas para tratar esta situación. En tales casos, se utilizan transformaciones no lineales del espacio de características o se emplean trucos matemáticos, como el uso de funciones de kernel [Cristianini and Ricci, 2008] (lineal: $K(\mathbf{X_i}, \mathbf{X_j}) = (\mathbf{X_i}^T \mathbf{X_j})$, polinómica de orden p: $K(\mathbf{X_i}, \mathbf{X_j}) = (\mathbf{X_i}^T \mathbf{X_j} + 1)^p$, sigmoide para determinados valores de β_0 y β_1: $K(\mathbf{X_i}, \mathbf{X_j}) = \tanh(\beta_0 \mathbf{X_i}^T \mathbf{X_j} + \beta_1)$, entre otras), para encontrar hiperplanos óptimos en espacios de mayor dimensión donde los datos puedan ser separables. Esto permite que las máquinas de vectores soporte sean muy flexibles y efectivas en una amplia gama de problemas de clasificación.

Una de las estrategias para tratar con datos no linealmente separables es realizar una transformación sobre el espacio de atributos ($\mathbf{X}$). Considerando $\mathbf{X} \in \mathbb{R}^n$ es posible realizar una transformación ϕ, tal que $\mathbf{X} \to \phi(\mathbf{X})$, donde $\phi(\mathbf{X}) \in \mathbb{R}^N$, generalmente siendo $N \gg n$. Sin embargo, esta transformación debe cumplir la siguiente condición

$$K(\mathbf{X_i}, \mathbf{X_j}) = \phi(\mathbf{X_i}) \cdot \phi(\mathbf{X_j}) \tag{3.42}$$

donde $K(\mathbf{X_i}, \mathbf{X_j})$ representa una función Kernel. Estas funciones tienen la particularidad de que pueden expresarse como productos escalares en el nuevo espacio de características $\phi(\mathbf{X_i})$. Esta característica es fundamental en el modelo y se refleja en la Ecuación 3.40, lo que permite generalizar dicha ecuación.

Si el conjunto de datos D no es linealmente separable en el hiperespacio definido por los vectores x_i, es posible que sea linealmente separable en el hiperespacio de mayor dimensión definido por los vectores ϕ_i [Cristianini and Ricci, 2008]. Esta técnica de utilizar funciones kernel y transformaciones no lineales nos brinda una poderosa herramienta para tratar con problemas de clasificación que no son linealmente separables en el espacio original de atributos.

Si lo anterior no es suficiente, la definición de margen puede modificarse para ser menos restrictiva. En lugar de buscar un hiperplano que separe completamente las clases con un margen fijo, se puede permitir que algunos puntos estén dentro del margen o incluso en el lado incorrecto del hiperplano. Este enfoque se conoce como "margen suave" y permite que el modelo sea más flexible y se adapte mejor a conjuntos de datos que no son completamente separables.

En el caso del margen suave, la función de error se ajusta para penalizar los puntos mal clasificados y los puntos cercanos al hiperplano de decisión. Lo cual se puede expresar de la siguiente manera:

$$y_i(\mathbf{W} \cdot \mathbf{X_i} + b) - 1 \geqslant 1 \rightarrow y_i(\mathbf{W} \cdot \mathbf{X_i} + b) - 1 \geqslant 1 - \epsilon_i. \tag{3.43}$$

En esta modificación, se introduce la variable de holgura ϵ_i que permite que algunos puntos estén dentro del margen o incluso en el lado incorrecto del hiperplano. Esto lleva a un aumento en la cantidad de vectores soporte y hace que el modelo sea más flexible y capaz de adaptarse a conjuntos de datos que no son completamente separables.

Es importante destacar que tanto el enfoque del margen suave como el uso de funciones kernel permiten crear una nueva máquina de vectores soporte que es más versátil y adecuada para una amplia gama de problemas de clasificación, incluyendo aquellos que no son linealmente separables en el espacio original de atributos. Para más detalles sobre estos enfoques, se puede consultar el trabajo de [Cristianini and Ricci, 2008].

3.2.4. Perceptrón multicapa

El perceptrón multicapa (MLP, por sus siglas en inglés) es una red neuronal artificial (ANN, por sus siglas en inglés) formada por múltiples capas [Bergmeir and Benítez, 2012,Russell et al., 2004, Bishop, 2006]. Siendo una ANN un modelo matemático que simula el comportamiento de una red neuronal biológica, cuyo componente principal es la neurona. Esta neurona tiene una representación matemática conocida como perceptrón. En la Figura 3.4, se muestra un esquema general de un perceptrón, que comienza con la información recibida de una instancia del conjunto de datos D, $X = (x_1, ..., x_n)$. Esta información debe ser procesada por el perceptrón para evaluar su nivel de relevancia en la toma de decisiones, lo cual está representado por los pesos w_i asignados a cada una de las variables de entrada.

La determinación de cada uno de estos pesos dependerá de la interpretación que realice el perceptrón en su parte interna. Esta considera tanto el conjunto completo de información de entrada, es decir, la suma ponderada $\mathbf{X} \cdot \mathbf{W} = \sum_i^n x_i w_i$, como la respuesta a esa información. Esto se realiza mediante una función de activación f =f($\mathbf{X} \cdot \mathbf{W}$), que puede variar de problema a problema. Siendo la función de paso binario y la función sigmoide ejemplos de funciones de activación.

Dada la estructura del perceptrón, es posible observar que sus características principales son el punto de entrada, el punto de interpretación y el punto de salida. A partir de esta base, es factible crear configuraciones más complejas en forma de capas que respeten las características antes mencionadas. Estas capas se llaman capa de entrada, capas ocultas y capa de salida, lo cual da origen al nombre de perceptrón multicapa.

En la Figura 3.5, se presenta un ejemplo de un perceptrón multicapa, donde se pueden observar las capas que lo componen. La primera de ellas es la capa de entrada, que es similar al caso del perceptrón simple. Las capas intermedias corresponden a las capas ocultas, y cada una de ellas está conectada de manera análoga a la capa de entrada, ya que también existen pesos entre los elementos de cada capa oculta. Finalmente, es posible ofrecer múltiples respuestas a través de la capa de salida.

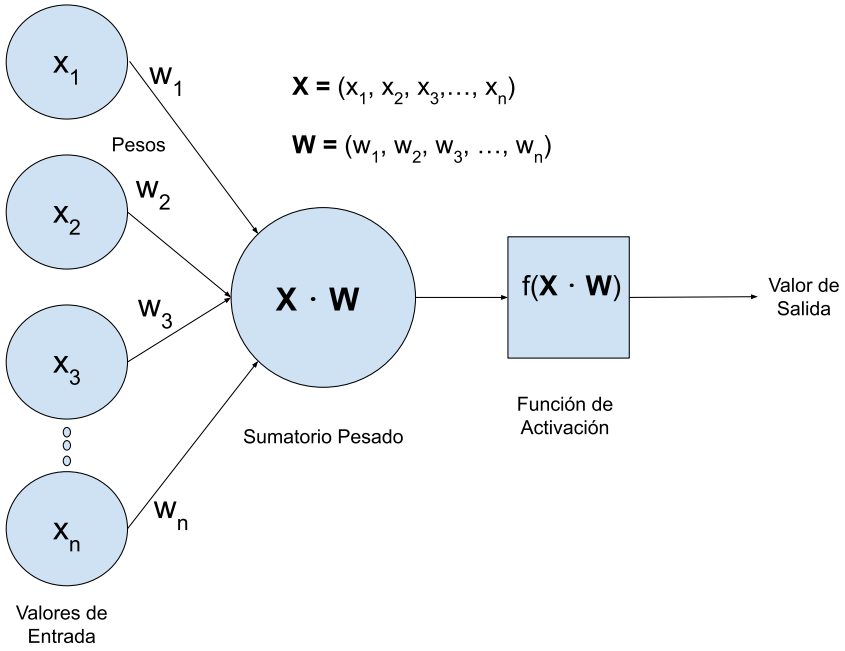

Figura 3.4. Representación gráfica de un perceptrón que recibe información del vector $\mathbf{X} = (x_1, x_2, ..., x_n)$, la cual tiene diferente grado de importancia considerado en los pesos, w_i. Vectores que serán utilizados para determinar el valor de la función de activación, f.

Ahora bien, hasta este punto solo se ha descrito la estructura de los perceptrones multicapa, pero no hemos mencionado cómo se determinan los pesos entre capa y capa, es decir, la interpretación del entorno D. Para lograr esto, se utilizan en conjunto el proceso de optimización del descenso del gradiente estocástico y la evaluación del error que ocurre entre capa y capa hasta llegar a la capa de salida [Bishop, 2006]. Este error debe propagarse desde la capa de salida hacia la capa de entrada, en un método conocido como propagación del error hacia atrás [Bishop, 2006]. Este proceso es esencial para ajustar los pesos de los perceptrones y hacer que la red neuronal aprenda de los datos proporcionados en el conjunto de entrenamiento. Con el tiempo y la repetición de este proceso durante el entrenamiento, la red neuronal es capaz de mejorar su rendimiento y aprender a realizar tareas de clasificación o regresión de manera más precisa.

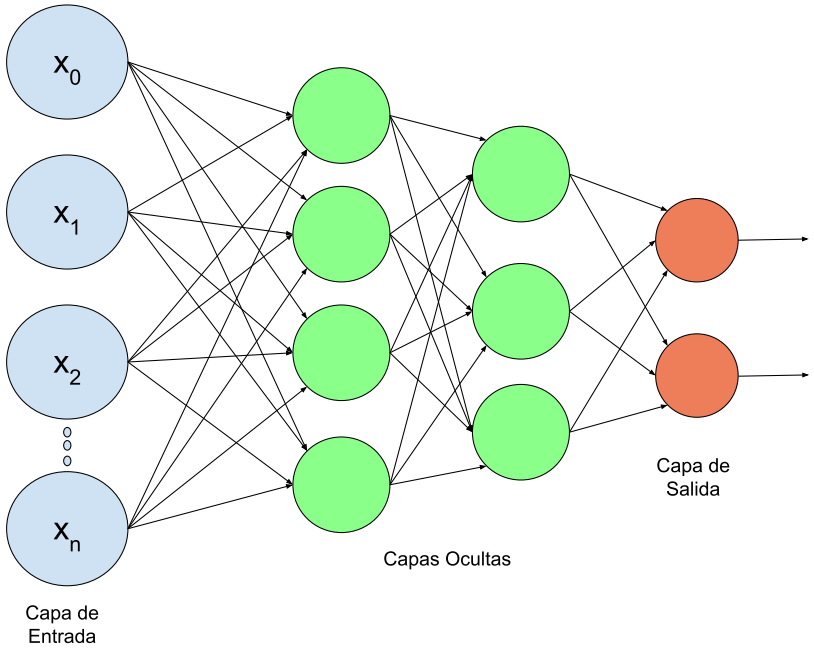

Figura 3.5. Representación gráfica de un perceptrón multicapa que recibe información del vector $\mathbf{X} = (x_1, x_2, ..., x_n)$, la cual es llamada capa de entrada. Dicha información será procesada por las capas intermedias, llamada capa oculta. Finalmente es capaz de ofrecer múltiples respuestas mediante la capa de salida.

Optimización

Considerando un perceptrón multicapa P con L capas, definimos P^i como la capa i-ésima del perceptrón, siendo P^1 y P^L la capa de entrada y salida, respectivamente. Los valores de entrada para la capa P^i estarán contenidos en el vector $\mathbf{a^{i-1}}$, donde $\mathbf{a^0} = \mathbf{X}$ corresponde al vector de entrada proveniente de una instancia del conjunto D, y $\mathbf{a}^L = \hat{\mathbf{y}}$ representa la respuesta del perceptrón multicapa.

Los pesos que conectan el perceptrón j de la capa l con el perceptrón i de la capa $l-1$ se representan como $w^l_{j,i}$. El objetivo es encontrar los pesos $w^l_{j,i}$ de tal manera que la función de error sea mínima [Russell et al., 2004,Bishop, 2006]. Para lograr esto, se utilizan los algoritmos del descenso del gradiente estocástico y la propagación del error hacia atrás.

Antes de continuar con la descripción de este proceso, es importante recordar que se está

trabajando en un problema de clasificación binaria. Por lo tanto, se podría utilizar la función sigmoide, al menos en la capa de salida, como función de activación en cada una de las capas del perceptrón multicapa. Esto se debe a que la naturaleza de la función sigmoide es resolver este tipo de problemas, como se observó en el modelo de regresión logística (Sección 3.2.1). Además, la función sigmoide ofrece un desarrollo estadístico apropiado, ya que sus valores están acotados en el rango de $|a_i| < 1$ y la derivada de la función está en términos de misma función. Esto es beneficioso para la clasificación, ya que se pueden interpretar las salidas como probabilidades que representan las pertenencias a una de las dos clases, facilitando la toma de decisiones.

Realizando un desarrollo similar al considerado en el modelo de regresión logística y utilizar el algoritmo del descenso del gradiente para optimizar los pesos $w^l_{j,i}$. En este contexto, la función de verosimilitud $\mathcal{L}$ (Ecuación 3.12) es conocida como función de costo o función de error E, por lo que el desarrollo matemático es análogo. Este proceso culmina con la actualización de los coeficientes (pesos). Resaltando la Ecuación 3.30, podemos reescribirla para este contexto como:

$$w^l_{j,i} = w^l_{j,i} - \eta \frac{\partial E}{\partial w^l_{j,i}} \tag{3.44}$$

donde η representa la tasa de aprendizaje, y $\frac{\partial E}{\partial w^l_{j,i}}$ es la derivada parcial de la función de error E respecto al peso $w^l_{j,i}$. La optimización de los pesos se realiza iterativamente mediante el descenso del gradiente, ajustando los valores de los pesos en cada iteración para minimizar la función de error E y lograr una mejor clasificación de los datos de entrenamiento.

El algoritmo de propagación del error hacia atrás es clave en este proceso, ya que permite calcular eficientemente las derivadas parciales necesarias para actualizar los pesos en cada capa. A través de este método, el perceptrón multicapa puede aprender de los datos de entrenamiento y ajustar sus pesos para mejorar su capacidad de clasificación en el problema dado. Este proceso se repite durante varias iteraciones hasta que la función de error converja a un mínimo y se obtenga un modelo de perceptrón multicapa bien ajustado a los datos de

entrenamiento.

Entonces, para observar la propagación del error hacia atrás se considera un perceptrón múltiple con solo una capa oculta y analizaremos el comportamiento de $\frac{\partial E}{\partial w^l_{j,i}}$ de la Ecuación 3.44 sobre una unidad ($w^l_{i,i}$ con i, j y l fijos) de la estructura.

Como se ha mencionado tenemos dos casos particulares (debido a que estamos considerando una capa oculta), si la unidad pertenece a la capa de salida o a la capa oculta. Sin embargo, independientemente de donde se encuentre la unidad debemos partir de una función de error (para una determinada instancia del conjunto de entrenamiento). Si estamos en la capa de salida, la función de error corresponde a $E = \frac{1}{2}\sum_i (y_i - a_i)^2$, donde hemos dejado el termino a_i y no $\hat{y}_i$ para observar la propagación del error.

Entonces, comenzando con la capa de salida ($l = L$) y considerando un peso específico calculamos $\frac{\partial E}{\partial w^L_{i,j}}$ resulta en

$$\frac{\partial E}{\partial w^L_{j,i}} = -(y_i - a_i)\frac{\partial a_i}{\partial w^L_{j,i}} = -(y_i - a_i)\frac{\partial f(\sum_j w^L_{j,i} a_j)}{\partial w^L_{j,i}} \tag{3.45}$$

donde $a_i = f(\sum_j w^L_{j,i} a_j)$ tal como se observa en la Figura 3.4. Lo cual puede ser reescrito como

$$\frac{\partial E}{\partial w^L_{j,i}} = -(y_i - a_i) f'(\sum_j w_{j,i} a_j)\frac{\partial \sum_j w^L_{j,i} a_j}{\partial w^L_{j,i}} \tag{3.46}$$

dando como resultado

$$\frac{\partial E}{\partial w^L_{j,i}} = -(y_i - a_i) f'(\sum_j w^L_{j,i} a_j) a_j \tag{3.47}$$

donde podemos observar que la actualización de la unidad $w^L_{j,i}$ se encuentra directamente relacionada con su correspondiente valor de activación de la unidad oculta, a_j.

Para la capa oculta se realiza un cálculo similar. Comenzando nuevamente con el análisis de error y observando que un cambio en $w^j_{k,j}$ se propaga hacia cada una de las unidades de la capa de salida. Por lo que se debe tomar en cuenta los errores de cada elemento en la capa de salida, resultado en la Ecuación 3.48

$$\frac{\partial E}{\partial w_{k,j}^{j}} = -\sum_{i}(y_i - a_i)\frac{\partial a_i}{\partial w_{k,j}^{j}} = -\sum_{i}(y_i - a_i)\frac{\partial f(\sum_j w_{j,i}^{j} a_j)}{\partial w_{k,j}^{L}} \tag{3.48}$$

donde no hay una relación directa entre $w_{k,j}^{j}$ y $w_{j,i}^{L}$ pero si con a_j, por lo que debe ser considerada en la derivada parcial

$$\frac{\partial E}{\partial w_{k,j}^{j}} = -\sum_{i}(y_i - a_i) f'(\sum_{j} w_{j,i}^{j} a_j)\frac{\partial \sum_j w_{j,i}^{j} a_j}{\partial w_{k,j}^{L}} \tag{3.49}$$

y dado que $w_{j,i}^{L}$ permanece constante frente a los cambios en $w_{k,j}^{j}$ que no estén relacionados con la componente j, tenemos lo siguiente

$$\frac{\partial E}{\partial w_{k,j}^{j}} = -\sum_{i}(y_i - a_i) f'(\sum_{j} w_{j,i}^{j} a_j) w_{j,i}^{j}\frac{\partial a_j}{\partial w_{k,j}^{L}} \tag{3.50}$$

donde tenemos que repetir el proceso de derivación pero ahora sobre a_j, el cual proviene de la capa de entrada (a_k) dando origen a la propagación hacia atrás del error. Siendo el resultado final

$$\frac{\partial E}{\partial w_{k,j}^{j}} = -\sum_{i}(y_i - a_i) f'(\sum_{j} w_{j,i}^{j} a_j) w_{j,i}^{j} f'(\sum_{k} w_{k,j}^{j} a_k) a_k. \tag{3.51}$$

Este resultado corresponde a un perceptrón multicapa sencillo pero este proceso puede para un perceptrón multicapa con más de una capa oculta. Para mayor detalle consultar [Russell et al., 2004, Bishop, 2006].

3.3. Índices de evaluación

Después de construir los diferentes clasificadores, es necesario aplicar indicadores de desempeño, que permitan medir sus rendimientos. Para ello, el estado del arte [Zou et al., 2007, Brown and Davis, 2006, Powers, 2020] ofrece diferentes opciones, como la exactitud, precisión, exhaustividad, puntuación F1, especificidad, entre otras, las cuales dependen del

umbral aplicado sobre la salida de los clasificadores. Es importante recordar que la salida de un clasificador, dado una instancia, ofrecerá generalmente un valor entre 0 y 1, dejando al usuario la elección del valor umbral para considerar si la instancia pertenece a la clase 1 o 0. Esta elección puede realizarse con la ayuda de la curva ROC. A continuación, describimos tanto la curva ROC como los diferentes índices antes mencionadas.

3.3.1. Curva ROC

De manera general, podemos observar que los clasificadores tienen una salida discreta, es decir, 0 o 1, ya que estamos considerando el caso binario. Sin embargo, estos valores están sujetos a una condición previa, que es el valor predeterminado de la probabilidad de que una instancia pertenezca a una clase o a la otra, generalmente este valor es P =0.5. Esta probabilidad puede observarse, por ejemplo, en el modelo de regresión logística (Ecuación 3.6), donde la función de probabilidad es $P(a) = \frac{1}{1+e^{-a}}$, es decir, una función continua con rango $(0, 1)$. Entonces, si $P(a) \geq 0{,}5$, se establece que el elemento o instancia pertenece a la clase 1, y de lo contrario, a la clase 0. De manera análoga, sucede en los demás clasificadores.

Entonces, la clasificación de un elemento $X \in A$ o $X \in B$ puede cambiar si el valor límite de P =0.5 es modificado. Por lo tanto, es importante analizar cómo varía la clasificación de X sobre los conjuntos A y B al variar el valor límite de P. Para esto, se utiliza la curva de Característica Operativa del Receptor (ROC, por sus siglas en inglés) [Zou et al., 2007, Brown and Davis, 2006, Powers, 2020]. Esta curva representa la proporción de instancias (X) clasificadas correctamente (clase 1), llamadas *verdaderos positivos*, contra la proporción de instancias clasificadas erróneamente, llamadas *falsos positivos*, respecto a la variación del valor umbral P. Un ejemplo de esta gráfica puede observarse en la Figura 3.6.

Una vez construida la gráfica ROC, se procede a calcular el área bajo la curva, donde un valor cercano a 1 indica que la cantidad de verdaderos positivos se aproxima al valor máximo y la cantidad de falsos positivos se aproxima al mínimo, lo que se cataloga como un buen clasificador.

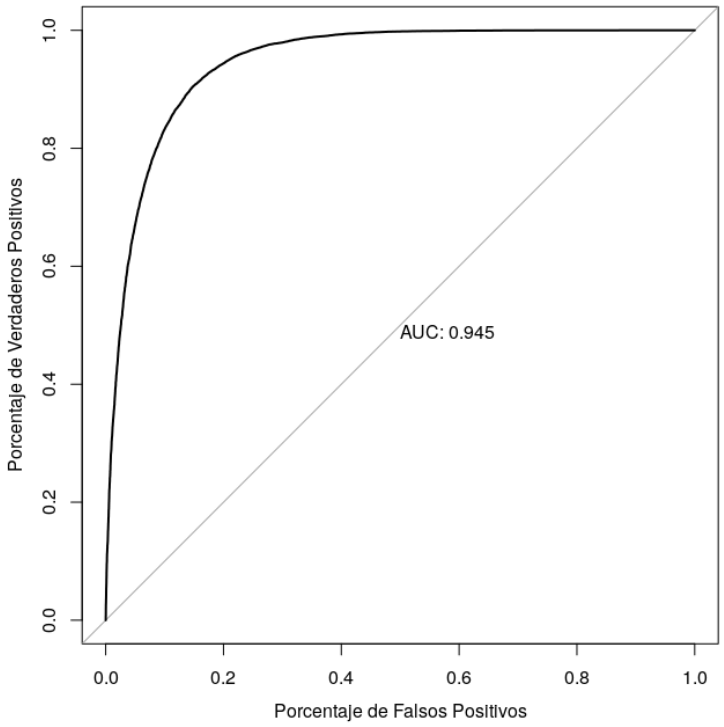

Figura 3.6. Ejemplo de Curva ROC, en el eje x se considera el porcentaje de Falsos Positivos y en el eje y el porcentaje de Verdaderos Positivos.

De esta manera, al construir la curva ROC y calcular el área bajo la curva para cada clasificador, se puede medir el rendimiento de cada uno de ellos y realizar una comparación directa entre todos aquellos construidos. Sin embargo, aún no hemos establecido el umbral de P. Generalmente, esta decisión se basa en las prioridades del problema a resolver, pero establecer índices a través de la matriz de confusión es de gran utilidad.

3.3.2. Matriz de confusión

Con la curva ROC construida y seleccionado el mejor clasificador (con mayor área bajo la curva ROC) es necesario establecer el valor del umbral P. Esto puede realizarse con los siguientes índices [Zou et al., 2007, Brown and Davis, 2006, Powers, 2020], la cuales pueden calcularse a partir de la matriz de confusión (Figura 3.7) compuesta de los siguientes elementos:

- **Verdaderos Positivos (VP)**: cuando el valor real del ejemplo es positivo y la respuesta

del clasificador es positiva.

- **Verdaderos Negativos (VN)**: cuando el valor real del ejemplo es negativo y la respuesta del clasificador es negativa.
- **Falsos Positivos (FP)**: cuando el valor real del ejemplo es negativo pero la respuesta del clasificador es positiva.
- **Falsos Negativos (FN)**: cuando el valor real del ejemplo es positivo pero respuesta del clasificador es negativa.

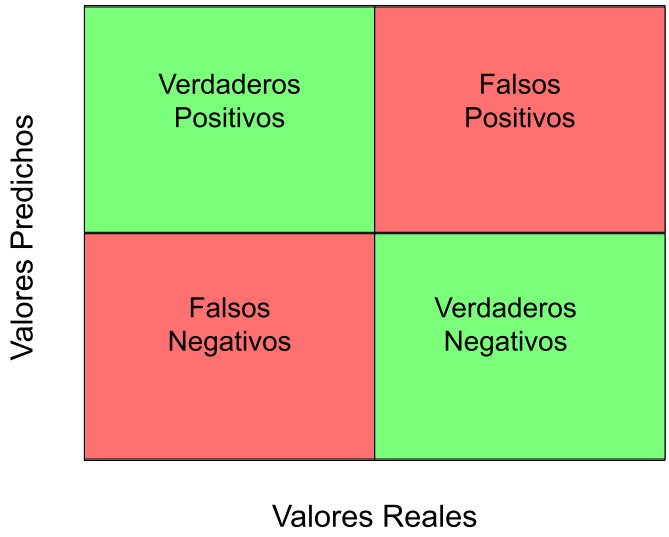

Figura 3.7. Matriz de Confusión.

A continuación se describe de manera breve los índices más utilizadas en el estado del arte: **Exactitud**, proporción de predicciones realizadas correctamente por el clasificador.

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN} \tag{3.52}$$

Precisión, proporción de predicciones positivas realizadas correctamente por el clasificador.

$$Presicion = \frac{VP}{VP + FP} \tag{3.53}$$

Sensibilidad, proporción de positivos reales correctamente identificados por el clasificador.

$$Sensibilidad = \frac{VP}{VP + FN} \tag{3.54}$$

Especificidad, proporción de negativos reales correctamente identificados por el clasificador (opuesto a la sensibilidad).

$$Especificidad = \frac{VN}{VN + FP} \tag{3.55}$$

Puntuación F1, es la media armónica ($H = \frac{N}{\sum_{i=0}^{N} 1/x_i}$) de precisión y sensibilidad. De manera general se puede considerar como una medida de la precisión y robustez del clasificador.

$$Puntuacion\, F1 = \frac{2VP}{2VP + FP + FN} \tag{3.56}$$

Capítulo 4

Arreglo experimental Belle II

Para conocer el contexto del problema a resolver y los datos a analizar se describe brevemente el experimento Belle II [Kou et al., 2018, Sagawa, 1996], el cual forma parte de la Organización Japonesa de Investigación de Aceleradores de Alta Energía (KEK, por sus siglas en japonés) y es el encargado de analizar las características de las colisiones de partículas que produce el acelerador SuperKEKB [Sup,]. Dicho acelerador de partículas se compone de dos anillos de una circunferencia de 3 Km, donde por un anillo viajan electrones y por el otro positrones. En un punto en particular ambos anillos se traslapan para producir colisiones entre electrones y positrones, cada una de estas colisiones es llamada *evento*. En este punto de colisión se encuentra el detector Belle II para estudiar los sucesos que allí ocurren. Una descripción gráfica del acelerador SuperKEKB y del detector Belle II se puede observar en la Figura 4.1.

Cabe mencionar que el detector Belle II está compuesto de los subdetectores: detector de pixeles (PXD), detector de silicio (SVD), contador (TOP), cámara central de acumulación (CDC), detector de kaones y muones (KLM), para medir las partículas cargadas e, π, K, μ, p y sus respectivas anti-partículas. Este tipo de partículas tiene la particularidad de dejar una rastro a través de los subdetectores en forma de trayectoria, la cual es llamada *traza*. También es capaz de medir las partículas no cargadas llamadas γ's, que a diferencia de las

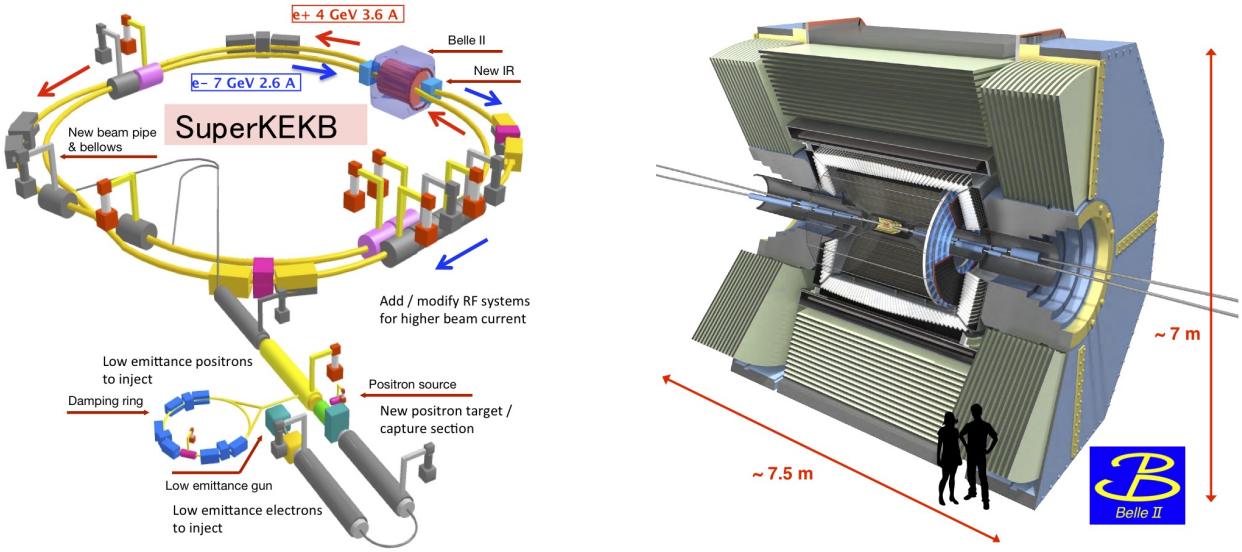

Figura 4.1. Arreglo experimental, acelerador SuperKEKB (izquierda) y detector Belle II (derecha) [Sagawa, 1996].

partículas cargadas no dejan un trayectoria, si no que depositan su energía de forma focalizada o cúmulos, llamados clusters. Este tipo de señales son medidas por el calorímetro (ECL). Sin embargo, a pesar de que el detector Belle II tiene la capacidad de observar las partículas cargadas antes mencionadas a través de sus trazas, no tiene la capacidad de reconocer a que partícula corresponde cada traza. Problema que es conocido como identificación de partículas y busca solución considerando el conocimiento adquirido en otros experimentos sobre las partículas y las características propias del detector, del tal manera que es posible asociar una probabilidad a cada traza de pertenecer a las partículas cargadas, para más detalles consultar [Kou et al., 2018, Sagawa, 1996].

Un ejemplo de este tipo de objetos llamados trazas o clusters se encuentra en la Figura 4.2, donde básicamente las trazas aparecen desde el centro del detector formando trayectorias, una linea continua. Por su parte, los clusters son aquellas señales que aparecen de manera aisladas. Este trabajo de identificación de partículas es de gran importancia para el experimento y para nuestro análisis y en el cual se ha hecho uso de herramientas de aprendizaje automático, pero que quedan fuera del alcance de este trabajo. Para mayor información consultar [McCormack

and Ganai, 2019, Esmail et al., 2019].

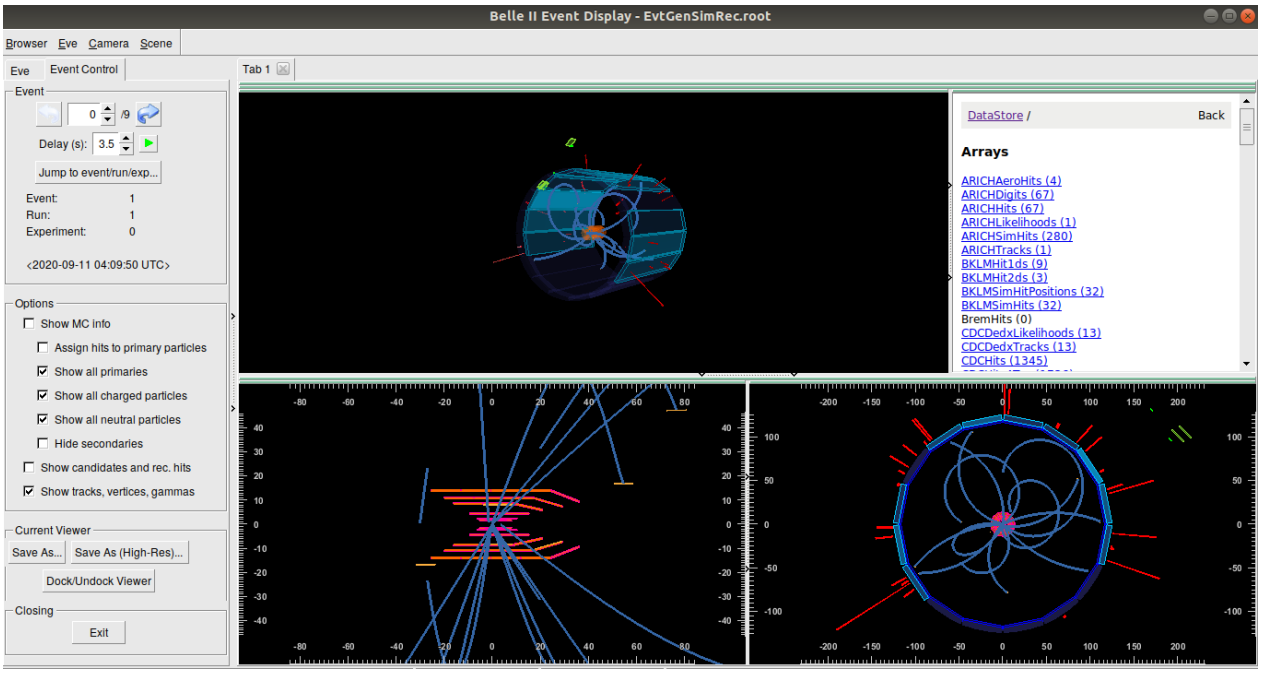

Figura 4.2. Ejemplo de trazas y clusters después de la reconstrucción del detector [Moll, 2011].

Esta información sobre las trazas y clusters será almacenada en los elementos de conjunto D, es decir, en X. Recordando que estamos estudiando el momento después de la colisión (evento), tendremos que X contendrá la información del conjunto de trazas y clusters reconstruidos después de dicha colisión. Ahora bien, el número de trazas y clusters no tiene que ser el mismo en cada evento. Por lo que para poder fijar un número determinado de variables independientes para X, es necesario considerar casos de estudio específicos. En este trabajo el decaimiento a estudiar es $\tau^- \rightarrow \pi^+\mu^-\mu^-\nu_\tau$.

Además, existe una complejidad adicional ya que al no poder establecer de manera determinista la pertenencia de una traza a un partícula específica por parte del detector, la información o cantidad de instancias por cada evento que produce el experimento se convierte en un problema combinatorio. Que dentro del punto de vista de las técnicas de aprendizaje automático no implica ninguna diferencia operativa para la construcción de los clasificadores, si no, en la necesidad de clasificadores más eficientes al encontrarnos en un problema de

desbalance de clases.

En resumen, el conjunto A contendrá a lo más una instancia por evento, la cual será seleccionada del conjunto de posibles combinaciones de trazas y partículas cargadas, el resto de ellas estarán en el conjunto B. Esto ocurre independiente de si trabajamos con datos medidos directamente del experimento o datos generados artificialmente. Sin embargo, en el caso de los datos proporcionados por el experimento no conocemos directamente que elemento corresponde al conjunto A o al conjunto B. Por lo que se opta por utilizar datos artificiales para entrenar las diferentes técnicas de aprendizaje automático.

4.1. Generación de datos artificiales

En la sección anterior (Sección 4) se mencionó de manera general las características de los datos proporcionados por el detector Belle II, los cuales se miden directamente del experimento. Sin embargo, esta clase de experimentos realiza estudios previos antes de iniciar su construcción tomando en cuenta los resultados que se esperan obtener. Parte de estos estudios son la generación de datos artificiales utilizando el muestas Monte Carlo [Jadach and Ward, 2000, Moll, 2011], las cuales simulan de manera virtual las características del acelerador, las capacidades del detector y la física que podría ocurrir durante las colisiones. Esto con el fin de no esperar hasta la toma de datos para iniciar el análisis y aún más importante para comparar la teoría (aquella aplicada en la generación de los datos artificiales) con los datos medidos esperados. Por lo que nosotros seguiremos el mismo camino para dar relevancia al decaimiento de estamos promoviendo.

En la Figura 4.3 se puede observar un diagrama general de lo que ocurre durante y después de una colisión, por el momento sin considerar el detector. Comenzando con la colisión electrón-positrón, seguida de todos los procesos físicos (recuadro verde) posibles, de los cuales consideramos que solo uno puede ocurrir por colisión. Estos procesos físicos son el principal motivo de estudio del experimento, más sin embargo, no pueden ser medidos directamente

por el detector. Para inferirlos se emplean las partículas finales (óvalo azul) que si pueden ser medidas por el detector.

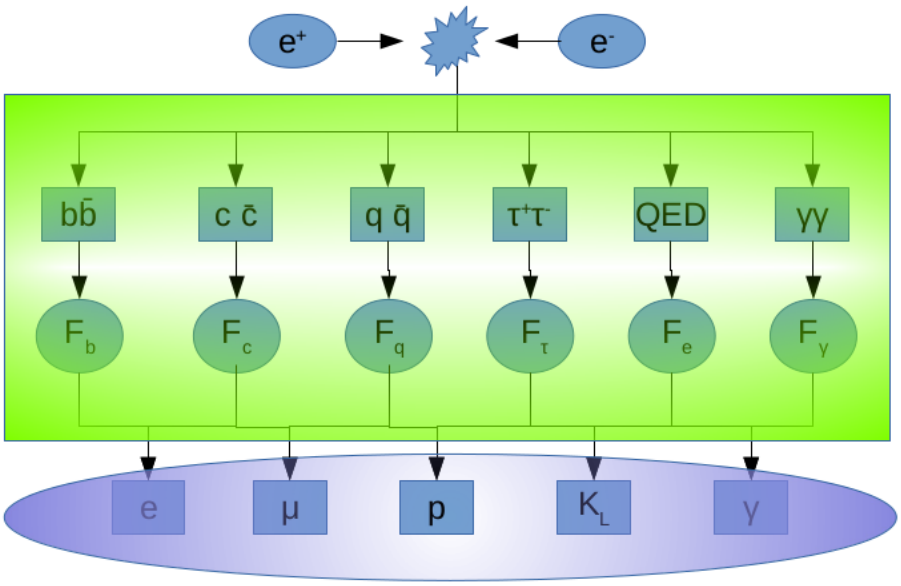

Figura 4.3. Diagrama de un evento en el cual no se consideran el detector Belle II y donde el flujo de información va de arriba hacia abajo.

Ahora bien, la descripción anterior de los eventos (colisiones) es independientemente si el detector Belle II está presente o no. Pero si consideremos este detector se tendrá más información disponible, es decir, las trazas y clusters producidos por las partículas finales. Entonces, considerando el uso del detector, representado en la Figura 4.4, es posible observar el flujo de información que utilizaremos en el presente trabajo. Comenzando desde la parte inferior con la información proporcionada por el detector hasta la parte superior (recuadro verde) que son los procesos físicos que se desean inferir.

Entonces, este proceso que ocurre dentro del experimento Belle II se simula de manera artificial utilizando técnicas de Monte Carlo. Con la ventaja de que es posible tener control los procesos físicos de interés (recuadro verde en las Figura 4.4). De esta manera se genera un muestra del conjunto A, que corresponde al decaimiento $\tau^- \to \mu^- \nu_\mu \nu_\tau$, simulando la colisión,

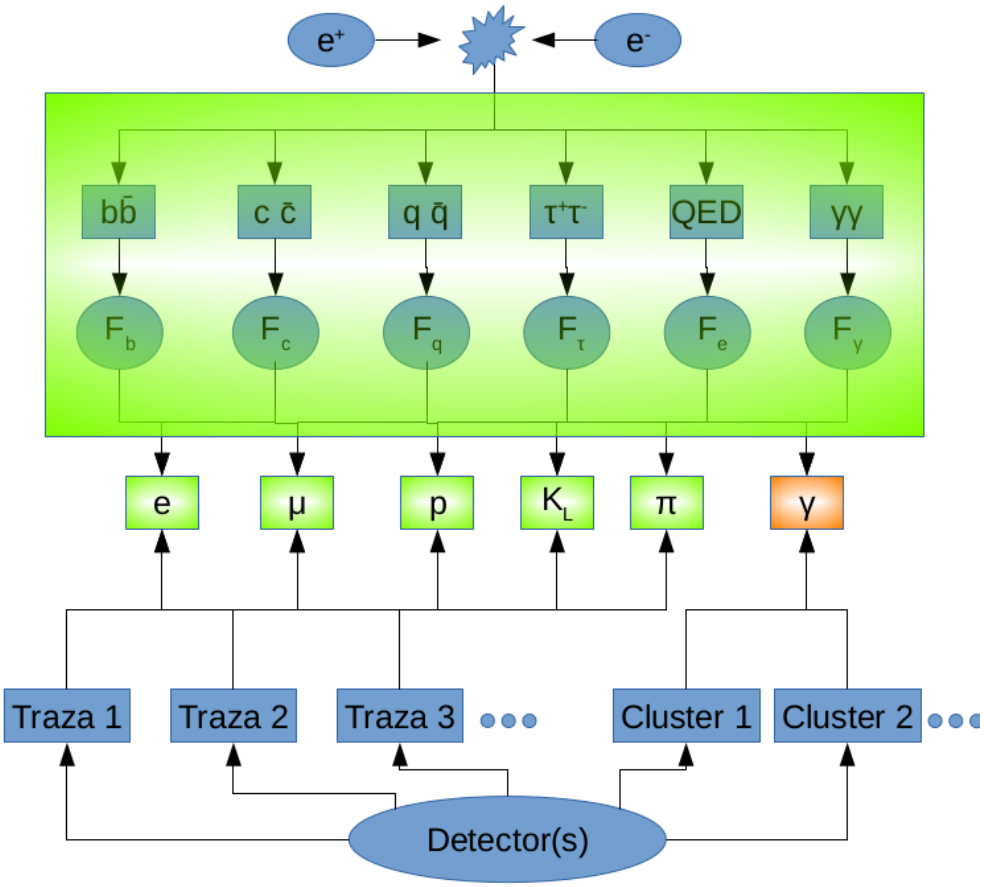

Figura 4.4. Diagrama de un evento en el cual se consideran el detector Belle II y donde el flujo de información va de abajo hacia arriba.

el proceso físico y el paso de las partículas generadas a través del detector. Esto se repite para cada decaimiento y finalmente tendremos una muestra tanto del conjunto A y B, sin olvidar que en cada caso se generan todos los candidatos posibles por evento, es decir, el conjunto C.

4.1.1. Selección de la de muestra Monte Carlo

Teniendo en cuenta el proceso de colisión de partículas y los datos que se generan, se puede decir que el interés final del experimento (recuadro verde de la Figura 4.3) es comparar la teoría de física de partículas con las mediciones hechas por el experimento y así lograr determinar si la teoría actual es correcta o si necesita ser modificada. En este sentido, este trabajo se

centra en una parte pequeña pero fundamental de la teoría de física de partículas [Quintero, 2014, López Castro and Quintero, 2013], el cual corresponde al decaimiento $\tau^- \to \pi^+\mu^-\mu^-\nu_\tau$, decaimiento que no ha sido medido actualmente por ningún experimento y que supone un gran reto debido a sus posibles fracciones de decaimiento [et al. (Particle Data Group), 2014, Quintero, 2014]. Entonces, al no existir un estudio previo de este proceso se optó por iniciar el estudio con un decaimiento que tuviese características similares que fuese a ser medido en los próximos años dentro del experimento Belle II. Esto permitirá hacer una comparación entre las técnicas que se implementaron y las técnicas que están siendo usadas actualmente por la colaboración y posteriormente puedan ser aplicadas no solo en el decaimiento en el cual se está interesado si no también en otros estudios, esto siempre y cuando se logre un resultado favorable.

Entonces, dentro del experimento se tiene que el decaimiento con características similares, es decir, con el mismo número de partículas cargadas pero considerando piones (π) en lugar de muones (μ) en el estado final respecto al principal caso de estudio y próximo a ser medido es el decaimiento $\tau^- \to \pi^+\pi^-\pi^-\pi^0\nu_\tau$. Así, la comparación entre la teoría y las mediciones experimentales comienza con la visualización de la teoría mediante muestras Monte Carlo, en este caso de la generación de la muestra correspondiente a $\tau^+\tau^-$ de la siguiente manera, proceso que ocurre después de la colisión (Figura 4.3). Además, consideraremos que después de la producción del par de τ's, uno de ellos decae en el decaimiento mas probable, es decir, tendremos el siguiente par de decaimientos (sus conjugados también son considerados):

- $\tau^+ \to \mu^+\nu_\mu\nu_\tau$
- $\tau^- \to \pi^+\pi^-\pi^-\pi^0\nu_\tau$

Después, las partículas que se produjeron tendrán que viajar a través del detector, ser medidas y reconstruidas. En este punto, es cuando construimos o seleccionamos los datos a analizar, pues tenemos un número fijo de trazas y clusters a considerar y que corresponden a la partículas producidas por el par de τ's.

De manera general, los datos estarán compuestos de combinaciones de trazas correspondientes a cuatro partículas cargadas (un μ y tres π's) y dos clusters correspondientes a una partícula neutra (π^0), para ver más detalles de la estructura de los datos consultar Anexo A.

Recordando que el detector no es capaz de identificar completamente una traza, la combinaciones de trazas antes mencionadas genera el conjunto total de candidatos (C), del cual solo se está interesado en el subconjunto que llamaremos **Candidatos Verdaderos**, el cual corresponde a la correcta relación entre traza y cada una de las partículas mencionadas anteriormente. Por lo tanto, si al menos una de las relaciones no es correcta, dicha combinación estará en el conjunto que será llamado **Candidatos Falsos**.

Cabe mencionar que a pesar de que el enfoque de este trabajo es clasificar si un candidato es **Falso** o **Verdadero**, es importante mencionar que existe la posibilidad de que el detector no pueda reconstruir el candidato verdadero de cada evento ya que suele limitarse el número de candidatos por evento, pero que la metodología aquí expuesta puede ser aplicada a cualquier otro proceso de decaimiento dentro del experimento.

Además, la metodología aplicará técnicas de clasificación binaria supervisada con el objetivo de realizar dicha separar eficientemente las partículas, utilizando las características extraídas de los datos. Logrando una mejor comprensión y clasificación de los eventos durante una colisión de partículas, esperando contribuir al avance tanto de la física de partículas como al campo del aprendizaje automático.

Capítulo 5

Metodología

Con el fin de encontrar el clasificador con el mejor rendimiento posible y resolver el problema de clasificación binaria, se propone una metodología que se inicia con la búsqueda de un conjunto adecuado de variables. Esto es especialmente relevante en este caso, ya que se enfrenta a un conjunto de datos que cuenta con más de 150 variables. Dada la alta dimensionalidad del conjunto de datos, es crucial realizar una selección cuidadosa de variables que sean relevantes y aporten información significativa para la tarea de clasificación. Esto nos permitirá reducir la complejidad del modelo y evitar problemas asociados con el sobreajuste.

La metodología propuesta considerará diferentes enfoques de selección de variables, como métodos de filtrado y envoltura, con el objetivo de identificar el conjunto óptimo que maximice el rendimiento del clasificador. Al realizar esta selección de variables de manera adecuada, podremos mejorar la interpretación de los resultados y aumentar la eficacia y eficiencia del clasificador en la tarea de clasificación binaria.

Siendo la propuesta un nuevo método de selección de variables que toma como referencia el método de eliminación hacia atrás mencionado en la Sección 3.1.2. Este nuevo método será descrito en la Sección 5.1.1 pero desde este momento se puede inferir que será necesaria al menos una técnica de aprendizaje para desarrollarlo, ya que el método de eliminación hacia atrás así lo requiere. Sin embargo, es importante destacar que el propósito de esta selección de

variables no es limitar la elección de la técnica de aprendizaje. Por lo tanto, se describe una metodología general (Sección 5.1) que contempla una serie de consideraciones para lograr los resultados deseados, ya que como se mencionó en el Capítulo 3 no es factible aplicar todas las posibles variaciones del conjunto de datos y los hiperparámetros de las técnicas de aprendizaje, entre otros aspectos, debido a que esto conduciría a la construcción infinita de clasificadores.

Por otra parte, nuestro método de selección de variables tiene la capacidad de comparar el rendimiento de una gran cantidad de clasificadores, por lo que seleccionar un clasificador a partir de la propia selección del conjunto de variables es una decisión informada y que puede resultar adecuada. Este punto de vista será abordado en la Sección 5.1.1.

Finalmente, la selección del clasificador podrá realizarse a partir de considerar la metodología general y una de sus múltiples posibilidades, o de la construcción del clasificador respecto a un conjunto de variables previamente seleccionado. En cualquiera de los casos se propone evaluar el rendimiento de los clasificadores a partir del valor del área bajo la curva (ROC), que proporciona una medida global del desempeño del clasificador. Al utilizar esta métrica, se puede comparar y generalizar el rendimiento de los clasificadores de una manera consistente y efectiva. Esto permitirá seleccionar el clasificador más adecuado para el análisis de decaimientos en el contexto del experimento Belle II.

5.1. Metodología general

La metodología para clasificar un decaimiento dentro del experimento Belle II (durante este trabajo el proceso $\tau^- \to \pi^+\mu^-\mu^-\nu_\tau$) se muestra en la Figura 5.1, donde se puede observar que inicia con la adquisición de datos, los cuales son la representación del arreglo experimental mostrado en el Capítulo 4 (en este trabajo el conjunto de entrenamiento será generado mediante simulación Monte Carlo (MC) y reconstrucción del detector Belle II) y que considera todos aquellos procesos de decaimiento que pueden ocurrir después de la colisión e^+e^- en el punto de intersección del acelerador.

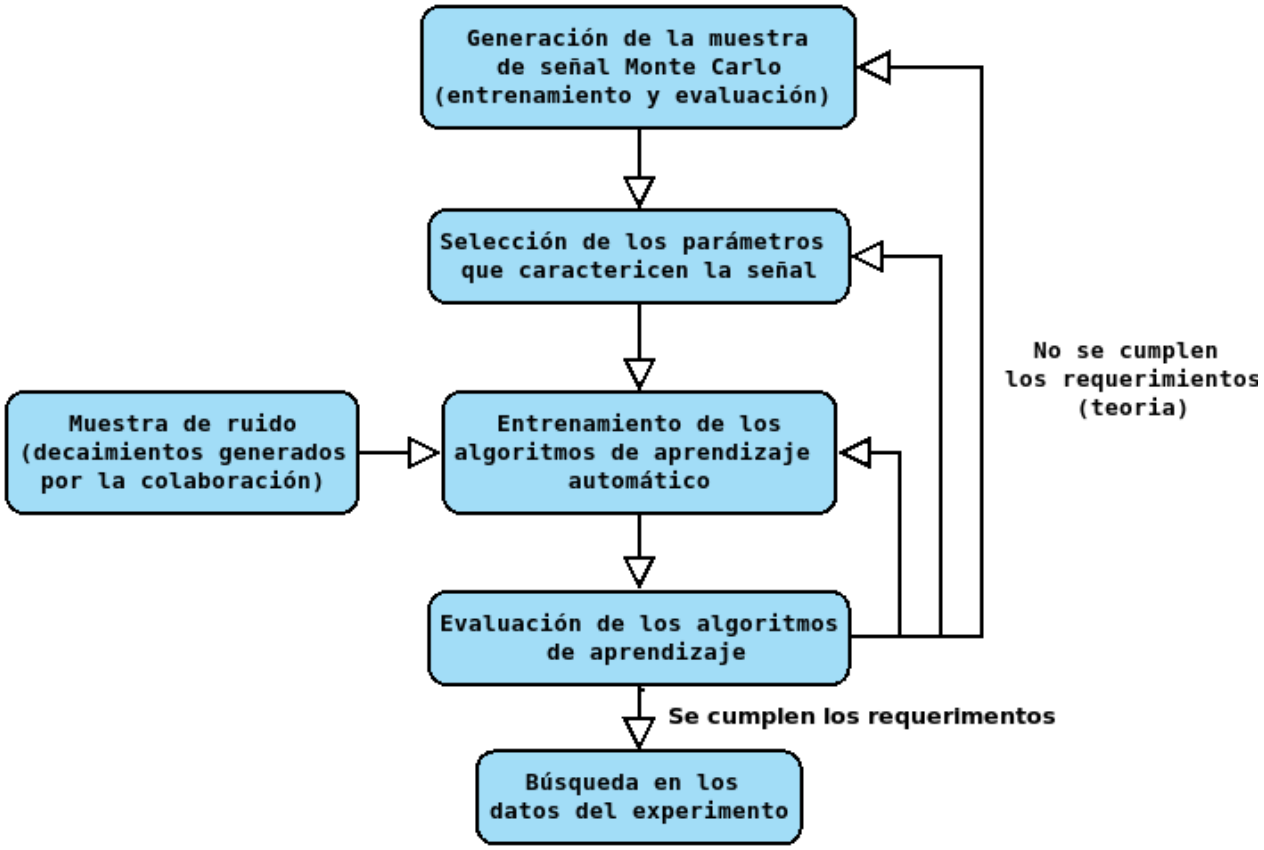

Figura 5.1. Metodología para clasificación de decaimientos en el experimento Belle II.

Durante el proceso anterior se generará un conjunto de datos D, el cual está compuesto por toda la información de generación, simulación y reconstrucción del detector. Siendo la información que realmente medirá el detector la que deberá ser usada para construir los diferentes clasificadores. Entonces, se seleccionarán aquellas variables que realmente medirá el experimento X y ya que estamos haciendo uso de información simulada vía MC agregaremos una variable de control Y para determinar a que tipo de proceso pertenecen las variables (evento o candidato). De esta manera se construirá el conjunto D y será lugar a un problema de clasificación binaria supervisada.

Antes de aplicar cualquier técnica de aprendizaje automático o estadística se recomienda hacer una observación sencilla sobre la calidad de los datos. Revisando que no existan registros vacíos en una instancia. De lo contrario tomar la decisión de eliminar la instancia

o complementar los registros con información. Siendo algunas opciones seleccionar un valor aleatorio o un valor promedio de esa registro respecto del conjunto D.

Una vez realizada la revisión previa es recomendable hacer otro estudio sobre el conjunto de variables o atributos $F = \{F_1, F_2, ..., F_n\}$ donde F_i representa al atributo i de las instancias $X \in D$. Lo cual puede llevar al proceso de selección de variables donde se puede utilizar alguno de los métodos de selección de variables descritos en la Sección 3.1. En nuestro caso desarrollamos un método híbrido que será descrito en la Sección 5.1.1. Pero independientemente del método de selección es posible trabajar con subconjunto $Q \subset F$ que representará a las variables relevantes del conjunto D.

A partir de este punto la construcción de diferentes clasificadores dependerá de las técnicas de aprendizaje automático que seleccione el usuario, así como de los hiperparámetros que utilice para inicializarlas.

Una vez finalizada la construcción de todos los clasificadores, continua la toma de decisiones. ¿Cuál es el mejor clasificador para nuestro problema?

Donde para responder esta pregunta proponemos hacer uso del área bajo la curva ROC. Ya que para el problema de clasificación binaria es una representación de la distribución de probabilidad de ambas clases, permitiendo la usuario ajustarse a las capacidades del clasificador. Esto utilizando los índices mencionadas en la Sección 3.3.

Ahora bien, que sucede si el rendimiento del clasificador no es suficiente para cumplir con los requerimiento del problema. Si esto ocurre tenemos diferentes opciones como se muestran en la Figura 5.1. La primera de ellas es aumentar el número instancias, lo cual es posible ya que dentro del experimento el entrenamiento de las técnicas de aprendizaje es con datos generador por MC, sin embargo, también aumentarán los tiempos de cómputo.

La siguiente opción es modificar los hiperparámetros de las técnicas de aprendizaje para hacer más o menos restrictiva la construcción de los clasificadores. Incluso modificar la estructura de la técnica como es el caso del perceptrón multicapa.

Finalmente, volver a realizar el estudio sobre el mejor conjunto de variables, Q, lo cual

también es posible dentro de nuestra propuesta de selección híbrida.

Todas estas consideraciones quedarán sujetas a las necesidades y recursos del usuario. Por el momento solo se concentran en la selección de variables y sus posibles modificaciones.

5.1.1. Metodología para la selección de variables

Como se mencionó en la Sección 3.1, seleccionar un buen conjunto de variables es un paso intermedio entre la determinación del conjunto D y la construcción de los diferentes clasificadores, llegando a ser un paso crucial debido a la cardinalidad que presentan ciertos conjuntos de datos y que puede dar lugar al problema de alta dimensionalidad, es decir, un posible sobreajuste en los parámetros de los clasificadores, lo que llevaría a una deficiente interpretación del entorno. Problemas que pueden considerarse independientes, tal y como ocurre con los métodos de selección por filtrado (Sección 3.1.1), sin embargo, el fin de cualquier clasificador es entregar el mejor rendimiento (definido de acuerdo a las necesidades de un problema específico) posible, por lo que optar por una metodología que estudie la relación entre el conjunto de datos D y el rendimiento de los clasificadores es interesante de analizar, tal y como lo realizan los métodos de selección de envoltura, envueltos e híbridos.

Entonces, al estar interesados en la influencia que tiene el conjunto de datos D sobre el rendimiento de los clasificadores, es necesario realizar una definición formal que permita medir esta influencia.

De manera general, el problema de selección de variables busca inferir un modelo matemático mediante un análisis sistemático sobre un conjunto de datos D, compuesto de las correspondientes variables predictivas X y la variable objetivo Y. Siendo el modelo matemático la función $f(\cdot)$, tal que, $f(X') \rightarrow Y'$, donde X' es un subconjunto de variables predictivas $(X' \subseteq X)$ y Y' es un vector objetivo, minimice la diferencia entre los vectores Y' y Y. Esto sobre el conjunto de prueba, ya que si se considera el conjunto de entrenamiento se podría construir un modelo de clasificación incorrecto. Donde el caso ideal se establece cuando $Y' = Y$, sin embargo, esto rara vez es ocurre sobre el conjunto de entrenamiento y que nos

llevará a definir cuándo un clasificador es *suficientemente bueno*.

Esta determinación de un buen clasificador no es un problema trivial, ya que cada caso de estudio puede establecer los requisitos mínimos para una buena solución, sin embargo, durante este trabajo proponemos (Sección 5.1.1) dar una opción general basada en el método de eliminación hacia atrás. Permitiéndonos establecer una relación entre un conjunto determinado de variables y el rendimiento de un clasificador construido de la siguiente manera.

Dado el conjunto $F = \{F_1, F_2, ..., F_n\}$ construiremos el subconjunto $Q = \{F_i, F_j, ..., F_q\}$ de acuerdo a la siguiente condición: $F_i \in F$ y $F_i \notin Q$ si F_i es redundante para el subconjunto Q. Donde establecemos que F_i es redundante si se considera en el proceso de construcción un modelo predictivo y la precisión del modelo resultante no supera la precisión del modelo construido sin F_i más allá un valor limite previamente establecido.

Ahora bien, teniendo en cuenta que se podrá establecer la rendimiento de un buen clasificador, también será necesario considerar cuáles y cuántas variables son relevantes para el modelo, es decir, el método para construir el conjunto Q. Para ello se hará uso de otro método de selección de variables, un método de filtrado que permitirá limitar el número de variables del conjunto Q.

Con esta breve descripción podemos establecer que la metodología será híbrida y que considera aprovechar las ventajas de los métodos de filtrado y envoltura. Metodología que será descrita con mayor detalle en la siguiente sección.

Método híbrido

Dada la descripción anterior del problema de selección de variables se retoma la necesidad de considerar el conjunto potencia del conjunto de variables, lo cual implica construir 2^n clasificadores tal y como se observó en la Sección 3.1, el método de eliminación hacia atrás es una aproximación que permite reducir el número de clasificadores a construir, un total de $n(n+1)/2$, seleccionando el mejor clasificador durante cada iteración, lo que significa que la decisión sobre el mejor conjunto de variables recae sobre n clasificadores, lo cual representa

una gran reducción sobre el espacio de clasificadores, pero también se tiene la posibilidad de seleccionar al clasificador construido con el conjunto total de variables F (clasificador seleccionado durante la primera iteración del método) y, por lo tanto, no se logre reducir la cardinalidad de dicho conjunto.

Esto no significa que no sea posible aprovechar la comparación que realiza de los diferentes clasificadores, de hecho, esto nos permite establecer un indicador sobre la eficiencia para un *buen* clasificador. Dado el clasificador resultante del método de selección hacia atrás, es posible calcular su respectiva área bajo curva ROC y establecerla como el punto de referencia para incluir o no una característica determinada F_j dentro del conjunto Q, lo cual puede interpretarse como la precisión que esperamos tenga un *buen* clasificador para el conjunto D.

Por otra parte, se desea alcanzar esta precisión con el menor número de características F_i, por lo que se opta por utilizar un método de filtrado, realizando un ordenamiento de sobre el conjunto de características F, de tal manera que sea generen subconjuntos con un número progresivo de características, respetando dicho orden para después construir los correspondientes clasificadores y seleccionar aquel que se aproxime a la referencia ofrecida por el método de selección hacia atrás utilizando el menor número de características y evitando el sobre ajuste causado por el gran número características disponibles.

El método de selección híbrido se resume en el Algoritmo 2, el cual requiere del conjunto D, el correspondiente conjunto de características F, un clasificador específico P y el parámetro h que nos permitirá ajustar el límite establecido para agregar o eliminar variables del subconjunto Q, el cual será el resultado final del algoritmo.

A continuación se realiza una descripción más detallada del Algoritmo 2 haciendo referencia a cada una de sus lineas: L1.- Una vez establecidas las condiciones iniciales del algoritmo se procede a aplicar el método de selección hacia atrás para obtener su correspondiente subconjunto de variables T. L2.- Calcular el valor del área bajo la curva (auc_T) del clasificador P respecto al conjunto T (referencia de eficiencia para un buen clasificador). L3-5.- Se calcula el valor de la información de cada uno de los los atributos del conjunto D. L6.- Se establece

Algorithm 2 Selección de Variable $\{Q\}$

Require: D, F, P, h
Ensure: $Q \subseteq F$
1: $T \Leftarrow$BEM(D, F, C) $//T \subseteq F$ es el conjunto de variables definido por el BEM
2: $auc_T \Leftarrow AUC(T, C, D)$
3: **for** $x_i \in S$ **do**
4: $iv_i \Leftarrow IV(x_i, D)$ // calcula el valor de la información de cada variable en S
5: **end for**
6: $X \Leftarrow Rank(x_i, iv_i)$ $//X$ es una secuencia de variables ordenadas por IV
7: $A_0 \Leftarrow \{\}$
8: **for** cada variable $x_i \in X$ **do**
9: x_i es la i-th variable en la secuencia X
10: $A_i \Leftarrow \{x_i\} \cup A_{i-1}$ // A_i es el conjunto que contiene las primeras i variables de X
11: **end for**
12: $i \Leftarrow 0$
13: **repeat**
14: $i++$
15: $auc_i \Leftarrow AUC(A_i, C, D)$
16: **until** $(auc_i \geq (h \times auc_T))$
17: $Q \Leftarrow A_i$

el ranqueo de los atributos. L7-11.- Creación de los conjuntos que serán utilizados para la construcción de los clasificadores. Iniciando con el conjunto A_1 que contiene la variable con el ranqueo más alto, seguido de A_2 con las dos variables con el ranqueo más alto, hasta el conjunto A_n que contiene todas las variables del ranqueo. L12-17.- Se utilizan los conjuntos A_i para construir los clasificadores usando la técnica de aprendizaje seleccionada y se calcula la correspondiente área bajo la curva ROC (auc_i). Esta construcción de clasificadores termina cuando se alcanzan las condiciones establecidas. En la cual se considera un valor de umbral h $(0 < h < 1)$ para considerar un error admisible respecto al punto de referencia (auc_T).

5.1.2. Selección del clasificador

La selección de un *buen* clasificador, como se ha mencionado generalmente se produciría después de seleccionar un *buen* conjunto de variables. Dicha selección de variables de manera general no tiene que depender de una técnica de aprendizaje automático, es decir, la selección de un conjunto de variables y de un clasificadores pueden considerarse como procesos inde-

pendientes. Sin embargo, debido a que durante nuestro proceso de selección de variables se construye una gran cantidad de clasificadores y además se busca al mejor de ellos de manera heurística, resulta en que el mejor clasificador se construye a partir del mejor conjunto de variables. Por lo que el proceso de selección del conjunto de variables y clasificador se realiza en un mismo paso.

Capítulo 6

Resultados experimentales

En este capítulo se muestran los resultados experimentales de la metodología presentada en el Capítulo 5, aplicada sobre una serie de conjuntos de datos, no solo sobre el conjunto de interés, ya que como se menciona en dicho capítulo nuestro proceso de selección se basa en una heurística. Por lo que antes de aplicar nuestra metodología debemos observar su comportamiento sobre conjuntos de datos independiente y ver sobre que condiciones es aplicable, recordando que una metodología o específicamente una técnica de aprendizaje automático no es capaz de resolver todos los problemas.

Este capítulo está divido en dos secciones principales: la primera, considera la metodología aplicada sobre los conjuntos de datos independientes a nuestro problema y la segunda, presenta la metodología aplicada sobre nuestro conjunto de interés pero tomando en cuenta algunas observaciones de los casos previos.

6.1. Resultados de conjuntos independientes

Antes de aplicar la metodología de selección (conjunto de variables y clasificador) a el caso de estudio se propone aplicarla sobre otros conjuntos datos independientes al trabajo, por lo que se eligieron cinco conjuntos de datos de libre acceso y que se encuentran disponibles en la plataforma Kaggle [kag, 2023]: *Boson de Higgs* [D, 2021], *Detección de fraudes en tarjetas de*

crédito [Dal Pozzolo et al., 2014], *Predicción de púlsares* [Raj, 2020], *Ataques cardiacos* [Sheta, 2021] y *Clasificación de datos bancarios* [Rashmi, 2020] . Este estudio nos permite evaluar de forma empírica el método de selección de variables, el cual corresponde a una propuesta heurística.

Los resultados presentados en esta sección corresponden a los siguientes conjuntos de datos: en el contexto de física se tomaron datos del bosón de Higgs [D, 2021] y predicción de púlsares [Raj, 2020]; en el contexto bancario se consideran la detección de fraudes en tarjetas de crédito [Dal Pozzolo et al., 2014] y suscripción a servicios crediticios [Rashmi, 2020]; dentro del área médica datos sobre ataques cardiacos [Sheta, 2021], donde se buscan los factores más relevantes de dicha enfermedad. Estos conjuntos de datos se describen a continuación con la misma nomenclatura del Capítulo 5, resaltado que a pesar de que algunos de ellos cuentan tanto con el conjunto de entrenamiento como de prueba solo se trabajará con el conjunto de entrenamiento, ya que no todos los conjuntos de prueba existe información sobre variable objetivo.

- *Boson de Higgs* [D, 2021]: este conjunto de datos es un proceso de física simulado que corresponde a las desintegraciones del bosón de Higgs y otros procesos secundarios. Estos datos se proporcionan en dos conjuntos de datos; un conjunto de entrenamiento T y un conjunto de prueba P, donde $T \cup P = D$ y $T \cap P = \{\}$. Los elementos de estos conjuntos de datos tienen 29 variables predictivas y una variable objetivo. También hay una variable de control que se omitió ya que solo hace referencia al número de instancias de los conjuntos. El conjunto de entrenamiento tiene 68,636 instancias, 36,279 instancias para procesos de Higgs e instancias para 32,357 procesos secundarios.

- *Detección de fraudes en tarjetas de crédito* [Dal Pozzolo et al., 2014]: este conjunto de datos contiene datos correspondientes a transacciones con tarjeta de crédito, cuyo objetivo es el de entrenar modelos para detectar transacciones fraudulentas. El conjunto de datos completo, D, tiene 284,807 observaciones y solo 492 de ellas corresponden a transacciones fraudulentas. Los datos originales de la transacción se han transformado

mediante el análisis de componentes principales. Los elementos de este conjunto de datos tienen 28 variables predictivas. El conjunto D, ha sido divido en dos subconjuntos: T (entrenamiento) y P (prueba), tales que $T \subseteq U$, $P \subseteq U$, $T \cup P = D$ y $T \cap P = \{\}$. Teniendo en cuenta que este conjunto de datos está muy desequilibrado ya que la clase positiva (fraudes) representa solo el $0.172\,\%$ de todas las transacciones.

- *Predicción de púlsares* [Raj, 2020]: este conjunto de datos contiene datos de candidatos a estrellas de neutrones que producen emisiones de radio detectables en la Tierra. Estos datos se proporcionan como dos conjuntos de datos: un conjunto de entrenamiento y un conjunto de prueba. Sin embargo, en este trabajo solo se considera el conjunto de datos de entrenamiento porque el conjunto de datos de prueba no contiene datos para la variable objetivo, es decir, D corresponde al conjunto de entrenamiento. Este conjunto de datos tiene ocho variables predictivas y una variable objetivo (binaria). Hay 9,273 observaciones en el conjunto de datos y 850 corresponden a pulsares. Este conjunto de datos D ha sido dividido en dos conjuntos: T (entrenamiento) y P (prueba), de modo que $T \subseteq D$, $P \subseteq D$ y $T \cup P = D$.

- *Ataques cardiacos* [Sheta, 2021]: este conjunto de datos contiene datos de 165 pacientes sanos y 138 pacientes con enfermedades del corazón. Hay 13 variables predictivas en el conjunto de datos y una variable objetivo (saludable frente a no saludable). Este es el conjunto de datos más pequeño que se usó en los experimentos y similarmente a lo que se ha descrito previamente para otros conjuntos de datos, se divide este conjunto de datos en un conjunto de entrenamiento T y un conjunto de prueba P, de modo que, $T \subseteq D$, $P \subseteq D$, $T \cup P = D$, y $T \cap P = \{\}$.

- *Clasificación de datos bancarios* [Rashmi, 2020]: estos datos están relacionados con campañas de mercadotecnia basadas en llamadas telefónicas de una institución bancaria portuguesa. El banco está interesado en saber si sus clientes suscribirían o no un producto financiero. Si bien hay dos conjuntos de datos disponibles, solo se considera el conjunto

de datos de entrenamiento, D, explícitamente. El número de clientes suscritos es de 3,712 y el número de clientes no suscritos es de 29,235, hay 20 variables predictivas en este conjunto de datos, además de la variable objetivo. De manera similar a lo que se ha hecho anteriormente, se divide este conjunto de datos en dos conjuntos; entrenamiento T y pruebas P.

Como se observa en la Tabla 6.1, estos conjuntos de datos no están balanceados, por lo que es necesario aplicar técnicas de balanceo y así mantener condiciones similares respecto a nuestros conjunto de datos.

Tabla 6.1. Descripción de los conjuntos de datos tomados de la base de datos *Kaggle* [kag, 2023].

Conjunto	Tipo	Clase 1	Clase 0	Instancias finales por clase
Boson Higgs	Entrenamiento	36,279	32,357	36,279
Fraudes Tarjetas	Total	492	284,807	284,807
Pulsares	Total	850	8,423	8,423
Ataques cardiacos	Total	165	138	165
Conjunto bancario	Entrenamiento	3,712	29,235	29,235

Conjuntos en los que la técnica SMOTE fue aplicada.

Para solucionar el problema de desbalance de clases se utilizó la técnica de sobremuestreo minoritario sintético (SMOTE, por sus siglas en ingles) [Chawla et al., 2002] para crear instancias sintéticas. Este método toma la clase minoritaria de la muestra y luego encuentra los k vecinos más cercanos para cada instancia, se elige un vecino aleatorio y se crea una nueva instancia en un punto intermedio entre ellos, esto en el espacio fase. El número de instancias generadas dependerá del factor de desequilibrio de cada conjunto de datos.

En estos casos de estudio se generaron las instancias necesarias para equilibrar las clases para los conjuntos de entrenamiento, no así para los conjuntos de prueba. Además, cabe mencionar que la técnica SMOTE no permite crear instancias sintéticas fuera del dominio de la clase minoritaria, por lo que si esta clase no es representativa podría causar un sobreajuste

en el clasificador.

Para mitigar el problema anterior se decidió realizar la validación cruzada de $k = 5$ iteraciones, es decir, dividiremos el conjunto en subconjuntos. Después, se utilizaran cuatro de ellos para el entrenamiento del clasificador y el quinto se utilizara para la evaluación del mismo. Proceso que se repetirá para cada uno de los cinco subconjunto de datos.

Por lo tanto el proceso de validación cruzada y la aplicación de la técnica SMOTE será la siguiente:

1.- Dividir cada conjunto D en los subconjuntos A y B, para dividir cada uno de ellos en cinco ($k = 5$) subconjuntos $A_1, ..., A_5$ y $B_1, ..., B_5$, respectivamente.

2.- De manera iterativa ($k = 1, ..., 5$) se construirán los siguiente conjuntos. Iniciando con $k = 1$, se dejará al conjunto $A_1 \cap B_1$ ($i = 1, .., 5$) como el conjunto de prueba D_P. El conjunto de entrenamiento será $A_T \cap B_T$, donde $A_T = A_2 \cap A_3 \cap A_4 \cap A_5$ y $B_T = B_2 \cap B_3 \cap B_4 \cap B_5$.

3.- Se continuará con la aplicación de la técnica SMOTE sobre el conjunto $A_T \cap B_T$, generando instancias del subconjunto con la menor cardinalidad (A_T o B_T).

4.- Se seleccionarán máx$\{|A_T|, |B_T|\}-$mín$\{|A_T|, |B_T|\}$ de instancias generadas para ser unidas con el conjunto de menor cardinalidad, tiendo conjuntos de cardinalidad máx$\{|A_T|, |B_T|\}$, los cuales serán llamados A_S y B_S.

4.- Finalmente $A_S \cap B_S$ será el nuevo conjunto de entrenamiento D_T.

En este punto ya se está listo para poner a prueba la metodología para la selección de variables y clasificador, por lo que se inicia aplicando el método de eliminación hacia atrás para tener un punto de referencia sobre el rendimiento de un buen clasificador. Para la aplicación de este método se consideró el modelo de regresión logística C, ya que es un modelo lineal y por lo tiene una complejidad temporal baja. Después, se realizó el ordenamiento de variables calculando el valor de la información, donde no solo se calculó el valor promedio del área bajo la curva ROC del método de eliminación hacia atrás (AUC(BEM)), sino que también del clasificador construido con el mismo número de variables del clasificador anterior pero con la variables del ordenamiento (AUC(IV)). Estos resultados se muestran en la Tabla 6.2, en la

cual se puede observar que el valor del área bajo la curva ROC en ambos casos no difiere de manera significativa. Por lo tanto, tomar uno u otro valor como referencia del rendimiento de un buen clasificador es posible. Sin embargo, no hay que perder de vista que los conjuntos de variables con los cuales se construyeron los clasificadores de referencia tienen la misma cardinalidad pero no necesariamente los mismos elementos.

Tabla 6.2. Área bajo la curva ROC (AUC) utilizando el método de eliminación hacia atrás y el valor de la información.

Conjunto	Variables	AUC (BEM)	AUC (IV)
Boson de Higgs	22	0.686	0.686
Detección de fraudes en tarjetas de crédito	27	0.997	0.998
Predicción de púlsares	7	0.978	0.977
Ataques cardiacos	9	0.950	0.953
Clasificación de datos bancarios	13	0.940	0.938

Valor de AUC para cada conjunto de variables de los conjuntos de entrenamiento del modelo de regresión logística.

Con los valores de rendimiento establecidos, se busca establecer la velocidad con la que aumenta el rendimiento de los clasificadores respecto al número de variables siguiendo el ranqueo establecido. Para observar este comportamiento y con fines ilustrativos se tomará como referencia al conjunto de datos del bosón de Higgs y una iteración de la validación cruzada. Entonces, se grafican los valores del área bajo la curva ROC de cada uno de los clasificadores respecto al numero de variables ranqueadas utilizadas. En la Figura 6.1 se muestran los resultados de los clasificadores de regresión logística contra el incremento de variables. Donde el número de variables seleccionado por el método de eliminación hacia atrás se representa con la linea punteada color rojo (linea vertical del lado derecho de la figura), el valor de referencia (auc_T) de dicho conjunto y correspondiente al clasificador de regresión logística se indica con la linea punteada color azul (linea horizontal) y el número de variables obtenida mediante la metodología propuesta se indica mediante la linea punteada color magenta (linea vertical del lado derecho de la Figura 6.1).

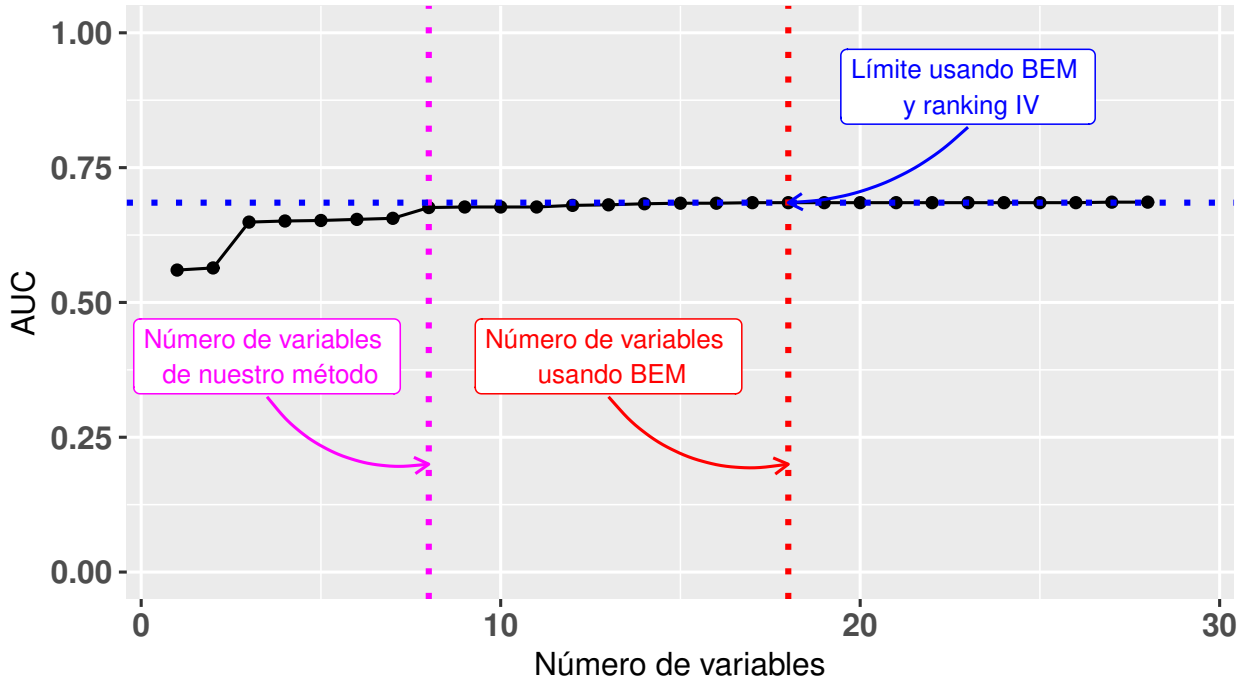

Figura 6.1. Valor del AUC de los clasificadores (LR) construidos incrementando el número de variables siguiendo el ranqueo por IV sobre el conjunto de entrenamiento. Conjunto de Higgs con un total de 28 variables por instancia.

Recordando que el usuario define el valor umbral h, para establecer una distancia entre el valor de referencia obtenido por el método de BEM y el valor del AUC a alcanzar, se tomo la decisión de seleccionar $h = 0.98$ después de haber realizado una serie de experimentos. Es decir, buscamos un conjunto de variables que permita construir un clasificador al menos 98 % tan bueno con el seleccionado por el método de eliminación hacia atrás, lo cual resulta en disminuir el número de variables de 21 a solo 8 variables, esto sin afectar significativamente la calidad del clasificador. Para tener mayor certeza de su buen rendimiento debemos aplicar los mismos clasificadores construidos sobre el conjunto de prueba. En la Figura 6.2 se puede observar el mismo comportamiento que en el conjunto de entrenamiento.

Ahora bien, lo anterior solo es válido para el modelo de regresión logística, por lo que también se realizaron estudios similares con las demás técnicas de aprendizaje automático. En particular se observa el comportamiento de la metodología respecto a las máquinas de

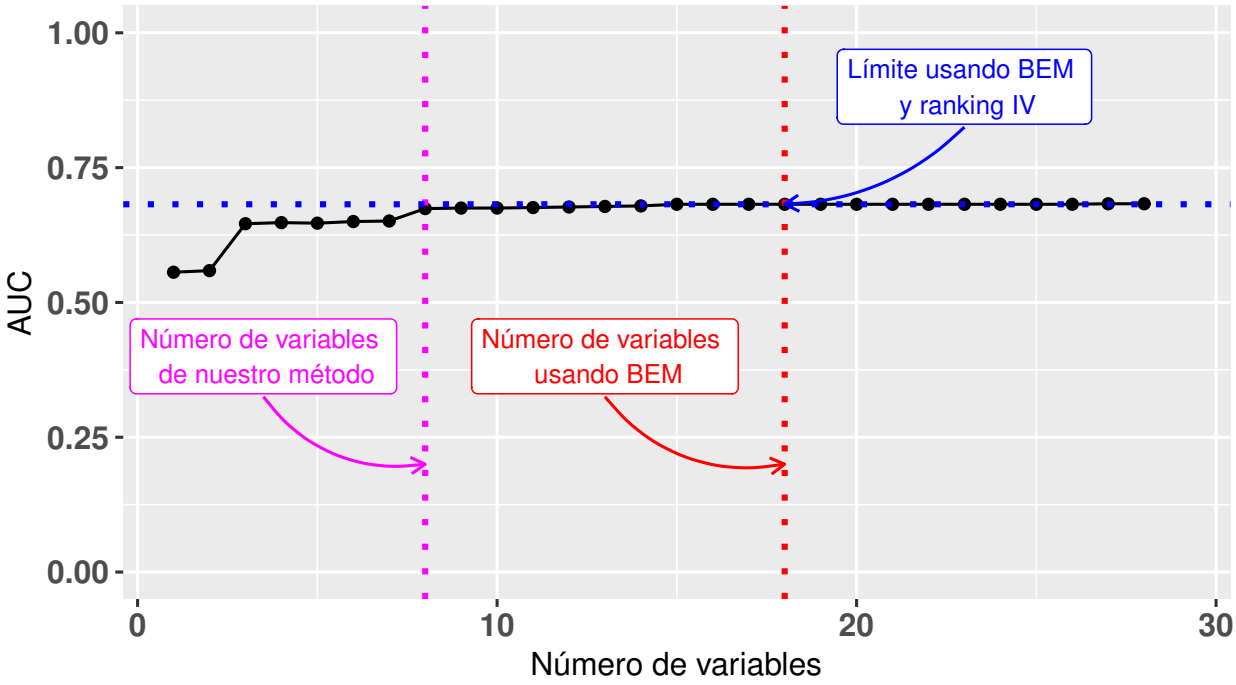

Figura 6.2. Valor del AUC de los clasificadores (LR) construidos incrementando el número de variables siguiendo el ranqueo por IV sobre el conjunto de prueba. Conjunto con un total de 28 variables por instancia.

vectores soporte, dado que en ellas no se limite el número de parámetros (número de vectores soporte) previo a su construcción, lo cual presenta una gran dificultad al momento de utilizar la metodología debido a los recursos de computo disponibles. Para solventar estas limitantes se toma como referencia el conjunto de variables obtenido al aplicar el método BEM con el modelo de regresión logística fijo ($BEM(C = LR)$). Con este conjunto de variables se construirá el clasificador utilizando cualquier técnica de aprendizaje y se establecerá una referencia (auc_T) aproximada.

Con esto en mente, se vuelve a aplicar la metodología (modificada) pero esta vez utilizando el modelo de máquinas de vectores soporte para la construcción de los clasificadores de acuerdo al ranqueo establecido (el cual es independiente de la técnica de aprendizaje a utilizar). En la Figura 6.3 podemos observar un comportamiento completamente diferente respecto al caso de regresión logística. Ya que no existe una tendencia del valor de AUC hacia un valor estable,

todo lo contrario, tiende a buscar el valor máximo 1. Esto es una señal de sobreajuste y se puede observar en la Figura 6.4, donde el valor del AUC alcanza un número reducido de variables para después disminuir su rendimiento, por lo que el resultado después de aplicar nuestra metodología no tendría ninguna relevancia sobre el modelo de máquinas de vectores soporte, sin embargo, tanto en la Figura 6.3 como en la Figura 6.4 hemos omitido la selección de variables respecto a esta técnica y nos enfocamos en la rendimiento de los conjuntos de variables obtenidos durante la aplicación del modelo de regresión logística (lineas punteadas verticales), donde se observa que a pesar de no dar un buen resultado para el conjunto de entrenamiento (Figura 6.3) si lo logra para el conjunto de prueba (Figura 6.4).

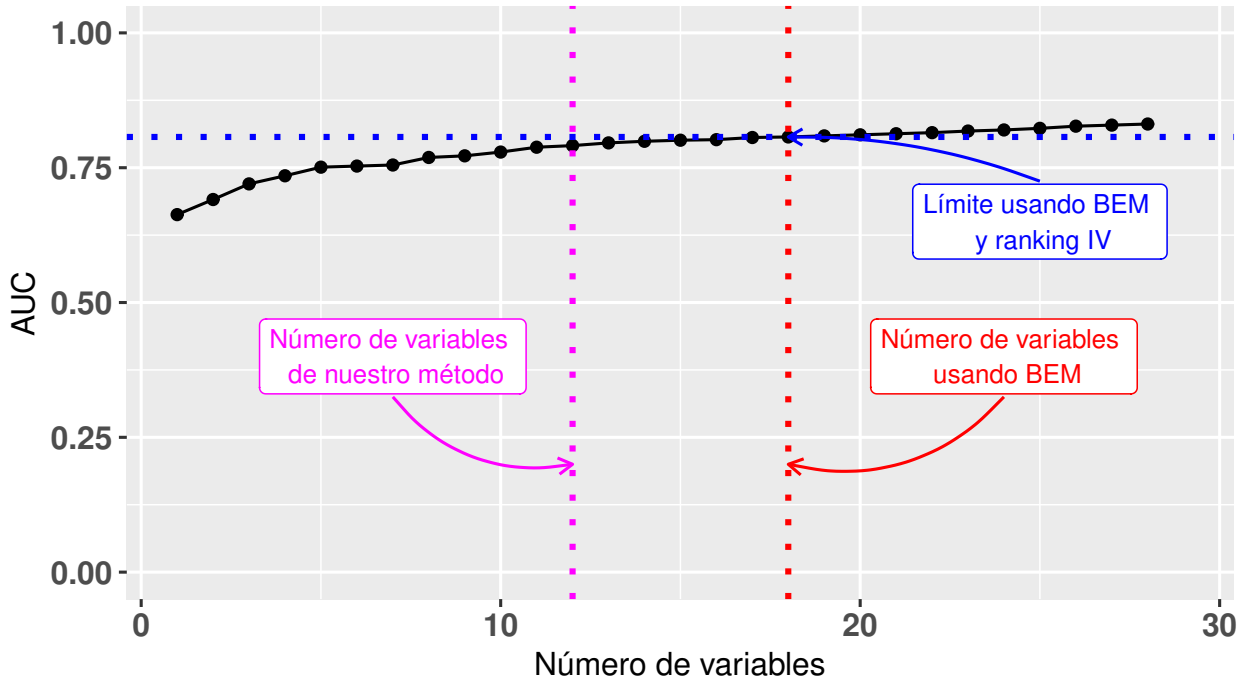

Figura 6.3. Valor del AUC de los clasificadores (SVM) construidos incrementando el número de variables siguiendo el ranqueo por IV sobre el conjunto de prueba. Conjunto Higgs con un total de 28 variables por instancia.

Para las técnicas de árbol de decisiones y perceptrón multicapa solo se toma referencia el número de variables proporcionado por el método BEM y el modelo de regresión logística para construir los clasificadores de DT y SVM y después calcular las respectivas áreas bajo la

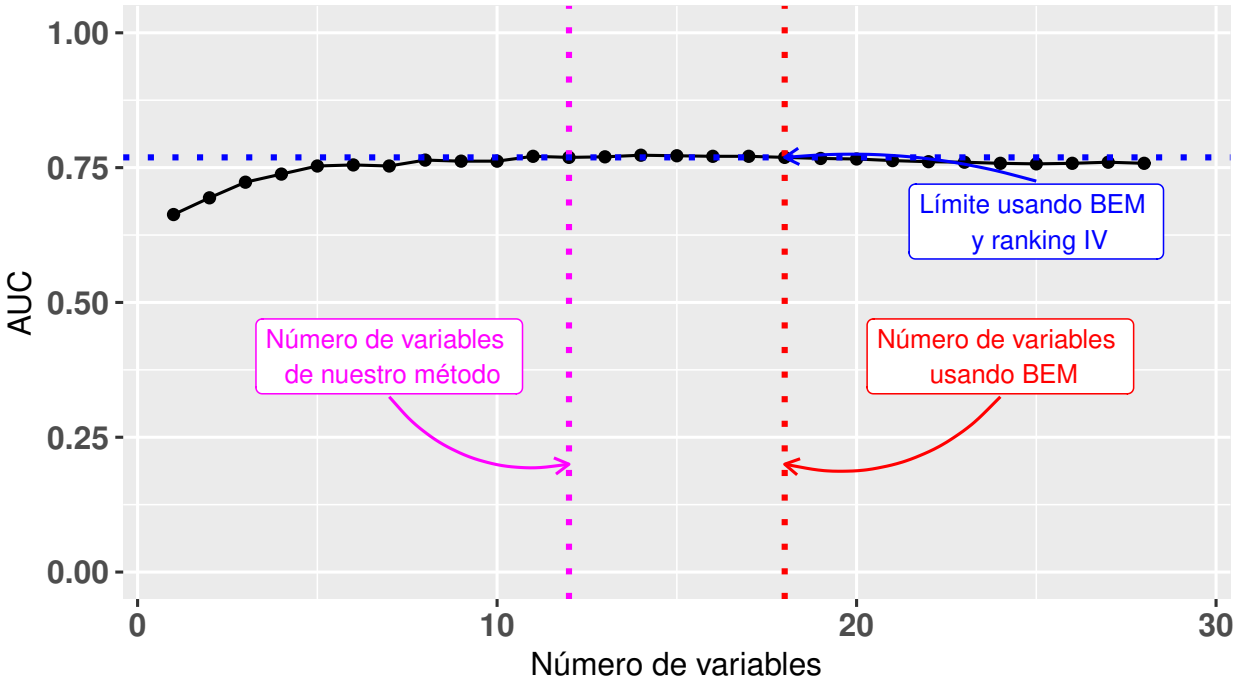

Figura 6.4. Valor del AUC de los clasificadores (SVM) construidos incrementando el número de variables siguiendo el ranqueo por IV sobre el conjunto de prueba. Conjunto Higgs con un total de 28 variables por instancia.

curva ROC. Los resultados obtenidos se resumen en la Tabla 6.3, donde se presentan los cinco conjuntos de datos independientes, el número de variables $|Q|$ después de aplicar nuestra metodología (modificada) sobre ellos y los valores de AUC (promedio de la validación cruzada, $k = 5$) tanto para los conjuntos de entrenamiento como para los conjuntos de prueba.

Este resultado muestra que no existe gran diferencia entre los valores de AUC de la curva ROC para los conjuntos de entrenamiento y de prueba, en cada una de las técnicas de aprendizaje automático considerada, es decir, estamos construyendo un buen clasificador para cada una de las técnicas y se puede observar su poder de clasificación dependiendo del conjunto a estudiar, permitiendonos la elección del más adecuado.

Con resultados favorables de la metodología para la selección de variables y clasificador, se procede a utilizarla en el principal caso de estudio: el decaimiento $\tau^- \to \pi^+\mu^-\mu^-\nu_\tau$.

Tabla 6.3. Número de variables ($|Q|$) y valor AUC para los conjuntos de entrenamiento y prueba, donde se consideraron los clasificadores: LR, DT, SVM and MLP.

Conjunto de datos		$\|Q\|$	AUC LR	$\|Q\|$	AUC DT	$\|Q\|$	AUC SVM	$\|Q\|$	AUC MLP
Bosón de Higgs	T	8	0.677	12.2	0.817	12.4	0.794	16.2	0.845
	P		0.673		0.736		0.766		0.762
Detección de fraudes en tarjetas de crédito	T	4.2	0.982	2.8	0.991	4.6	0.987	3.8	0.984
	P		0.953		0.923		0.953		0.953
Predicción de púlsares	T	1	0.974	2.4	0.976	1	0.969	1	0.974
	P		0.972		0.967		0.972		0.972
Ataques cardiacos	T	6	0.911	8	0.948	5.6	0.950	4.4	0.958
	P		0.882		0.843		0.911		0.838
Clasificación de datos bancarios	T	6.8	0.934	7.2	0.964	6.8	0.946	7	0.956
	P		0.896		0.897		0.916		0.898

6.2. Resultados de nuestro conjunto de datos

Como observamos en la sección anterior la metodología para la selección de un buen conjunto de variables así como de un buen clasificador, no presenta el mejor rendimiento para todas las técnicas de aprendizaje, lo cual no es de sorprender ya que como se ha mencionado no existe una única técnica que resuelva todos los problemas. Además, se vio la necesidad de hacer nuevas consideraciones debido a los recursos de cómputo disponible, sobre todo en el número de variables (conjunto de variables) que ofrece el método BEM, ya que de acuerdo al Algoritmo 2 durante todo el proceso se debería considerar el mismo clasificador (técnica de aprendizaje automático). Sin embargo, los recursos que exige el método BEM para construir $n(n+1)/2$ clasificadores depende de la complejidad de la técnica de aprendizaje, así como de la cardinalidad del conjunto D. En particular para los algoritmos de SVM y MLP su complejidad computacional es muy superior a la de RL. Entonces, a pesar de que la metodología puede mantener la misma técnica de aprendizaje si se dispone de los recursos necesarios, una buena aproximación es considerar el método BEM aplicado al modelo de regresión logística.

Para el conjunto de interés utilizamos la aproximación de la metodología antes mencionada (Algoritmo 2), ya que la cardinalidad del correspondiente conjunto F es 162. El resultado correspondiente al modelo de regresión logística se muestra en la Figura 7.1, donde la linea punteada roja representa el número de variables proporcionado por el método BEM, la linea punteada azul el valor del AUC del clasificador construido con las variables proporcionado por BEM y la linea punteada magenta el número de variables proporcionado por nuestra metodología. Esto para el conjunto de entrenamiento.

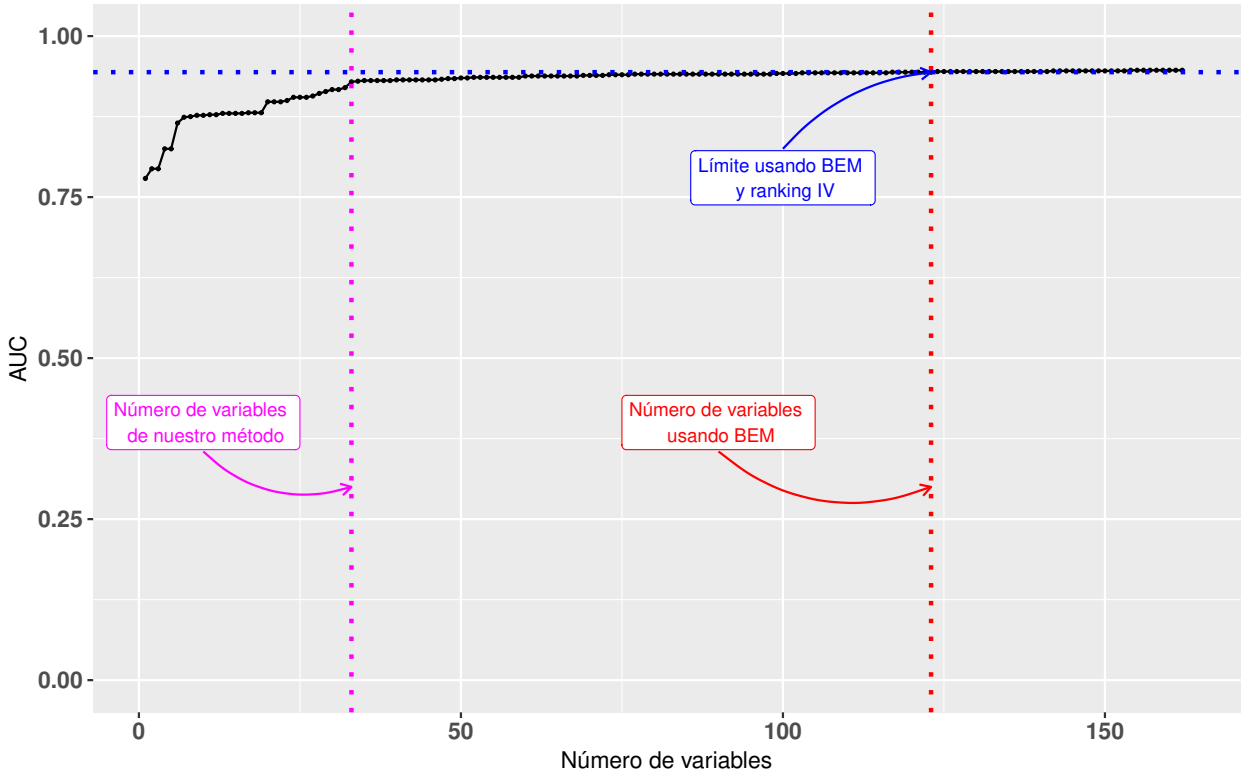

Figura 6.5. Variables seleccionadas utilizando el método BEM (linea roja) y nuestro método (linea magenta) para regresión logística (LR) sobre el conjutno de entrenamiento. El límite AUC queda determinado por el método BEM (linea azul).

En la Figura 7.2 se muestran los resultados para la técnica de árboles de decisión sobre el conjunto de entrenamiento utilizando la metodología modificada. A pesar de que la complejidad

para este modelo no representa un gran costo computacional se decidió mantener la referencia de BEM aplicada a RL para todos los casos de estudio, observando un comportamiento similar cuando se usa el modelo de RL, es decir, una tendencia a un valor constante sin alcanzar el valor máximo.

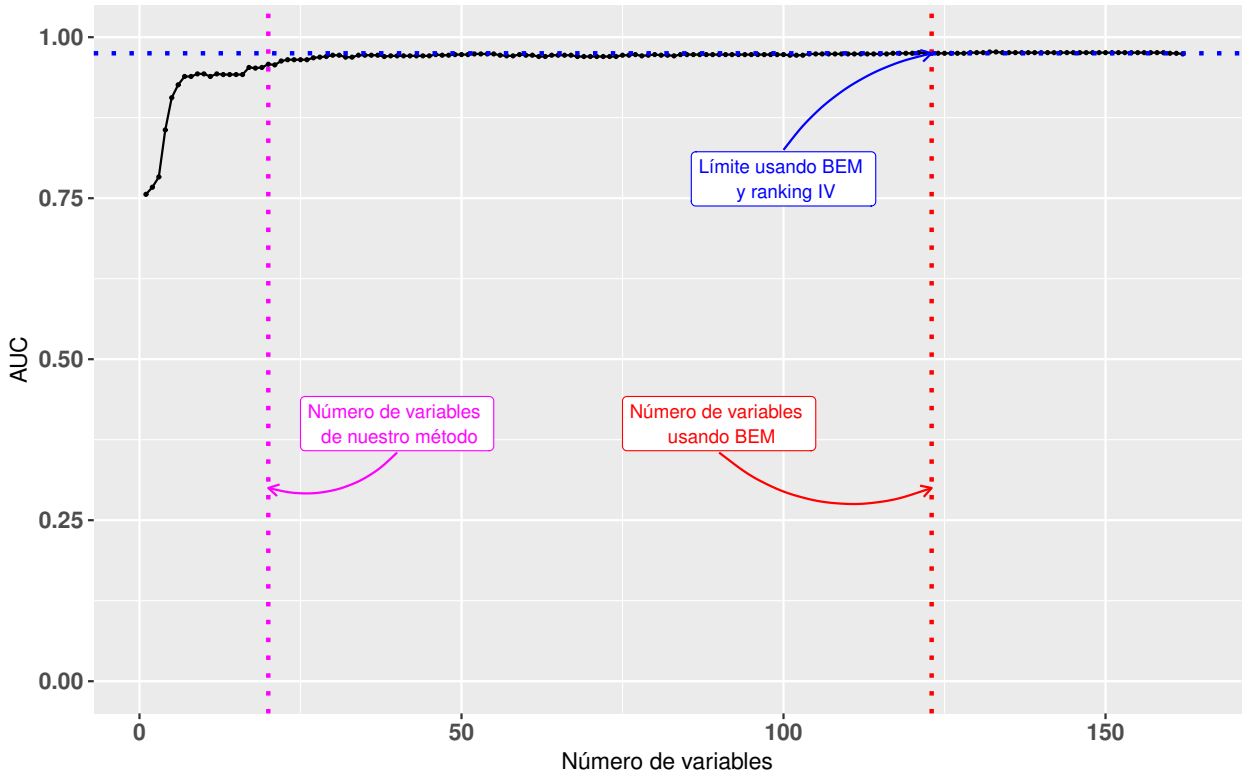

Figura 6.6. V
ariables seleccionadas por la metodología y DT, conjunto Higgs.Variables seleccionadas utilizando el método BEM (linea roja) y nuestro método (linea magenta) para árboles de decisión (DT) sobre el conjunto de entrenamiento. El límite AUC queda determinado por el método BEM (linea azul).

Para el caso de SVM no fue posible llevar a cabo el análisis, incluso con solo 162 clasificadores, debido a los limitados recursos de cómputo. Por lo que se limito la construcción a solo 40 clasificadores. En la Figura 6.7 se muestra solo el número de variables seleccionadas por la metodología de modelo de regresión logística. Recordando que debido a su tendencia de sobre

ajustar los clasificadores optamos por observar el comportamiento del conjunto de variables seleccionado por el modelo de RL.

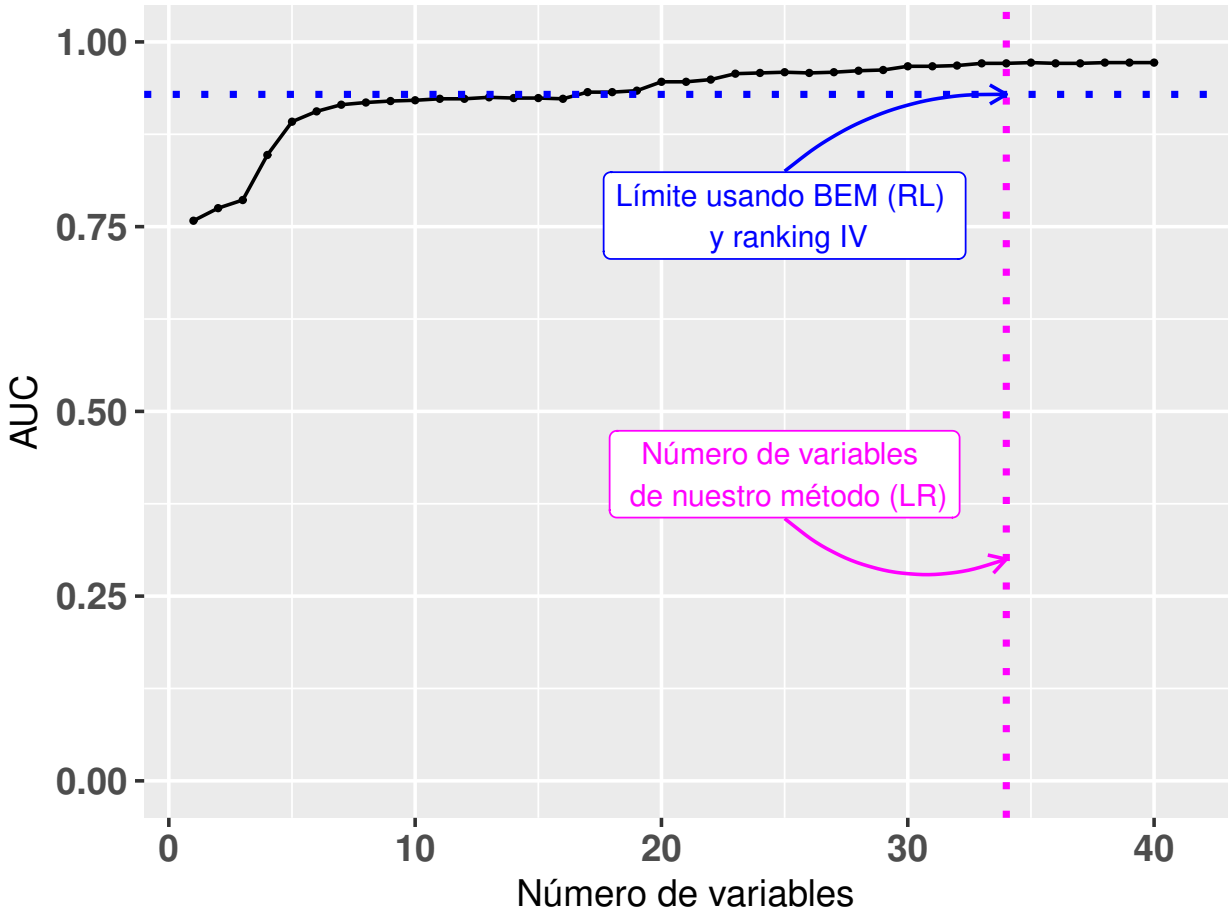

Figura 6.7. Variables seleccionadas utilizando nuestro método (linea magenta) con el regresión logística (LR) y el valor límite correspondiente del AUC dado por el método BEM (linea azul) sobre el conjunto de entrenamiento.

Finalmente, para el perceptrón multicapa ocurrió el mismo fenómeno que para SVM, la complejidad del algoritmo y los recursos de cómputo disponible hicieron imposible construir los 162 clasificadores para aplicar la metodología modificada, por lo que también se optó por observar el comportamiento del conjunto seleccionado por la metodología aplicada al modelo de RL. Los resultados obtenidos se muestran en la Figura 6.8, donde se delimita el rendimiento obtenido por el clasificador de RL y el respectivo número de variables. Cabe

mencionar que la estructura del perceptrón multicapa depende de la experiencia, es decir, en este caso se optó por mantener una relación 2 a 1 del número parámetros del perceptrón respecto número de variables de entrada, lo cual significa que la estructura del perceptrón y su complejidad va aumentando al incrementar el número de variables. Esto representa un serio problema cuando el número total de variables es considerable (162). Comportamiento que no se observó para los conjuntos independientes, ya que el número de variables no supera las 30 unidades.

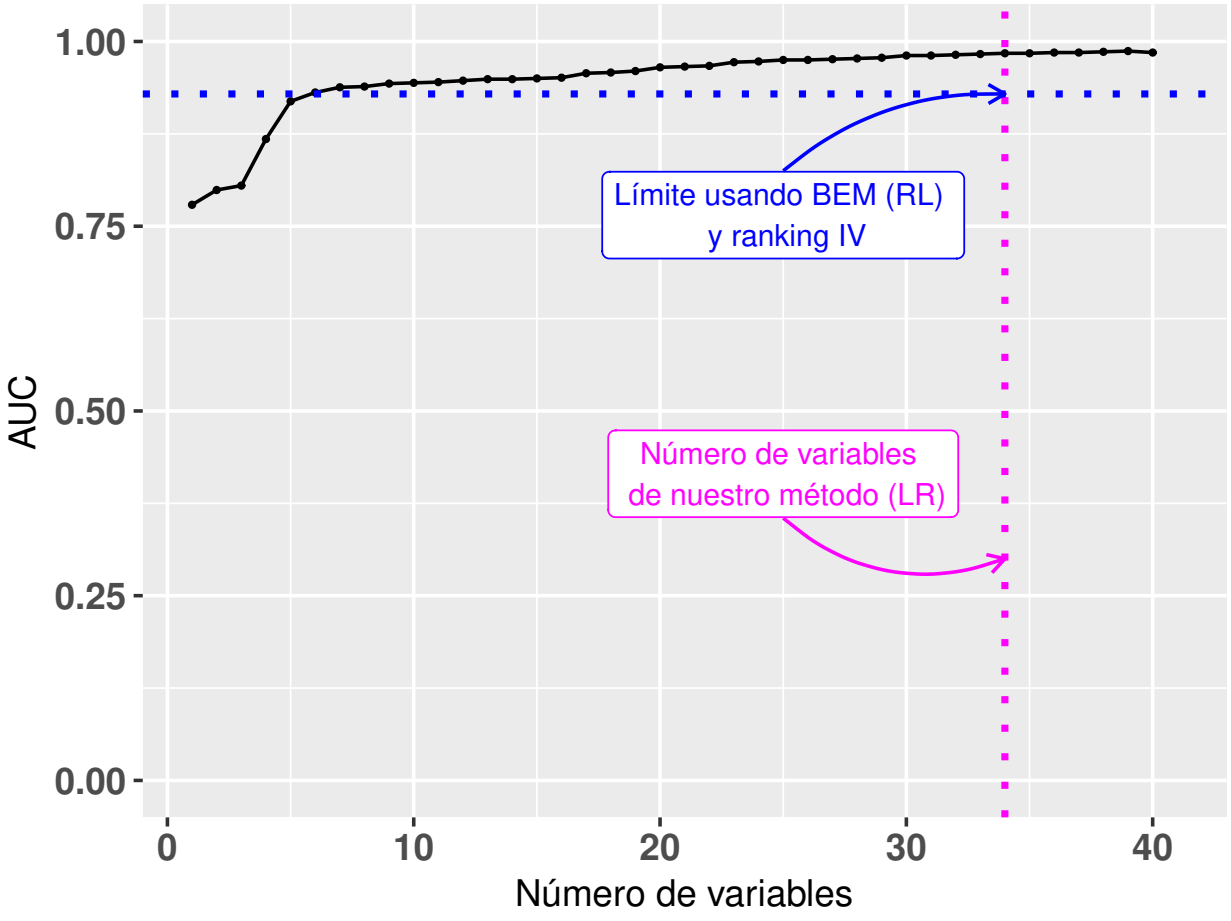

Figura 6.8. Variables seleccionadas utilizando nuestro método (linea magenta) con el regresión logística (LR) y el valor límite correspondiente del AUC dado por el método BEM (linea azul) sobre el conjunto de entrenamiento.

Los resultados obtenidos en los conjuntos de entrenamiento y los conjuntos de prueba se

resumen en la Tabla 6.4.

Tabla 6.4. Numero de variables ($|Q|$) y valor de AUC para los conjuntos de entrenamiento y prueba. Donde se consideraron los clasificadores: LR, DT, SVM y MLP.

Conjunto de datos		$\|Q\|$	AUC LR	$\|Q\|$	AUC DT	$\|Q\|$	AUC SVM	$\|Q\|$	AUC MLP
Decaimiento τ	Entrenamiento	4.2	0.982	2.8	0.991	4.6	0.987	3.8	0.984
	Prueba		0.953		0.923		0.953		0.953

Capítulo 7

Discusión y conclusiones

Con base en los experimentos realizados, se pueden observar varias características que surgen entorno a nuestra metodología. En primer lugar, se mencionan las observaciones sobre el método de selección de variables (clasificador) propuesto (Algoritmo 2) aplicado sobre los conjuntos de datos independientes. Luego, se destacan los resultados obtenidos en el caso de nuestro conjunto de datos específico y su aplicación en la física de partículas, teniendo en cuenta que cada entorno requiere un estudio específico. Finalmente, se presentan las conclusiones acerca de nuestra metodología.

7.1. Conjuntos independientes

Una de las características importantes que se quiere resaltar sobre los conjuntos independientes es el número de variables que determinan el conjunto F, tal y como se describe en la Sección 6.1, no supera las 30 unidades. Esta baja cardinalidad favorece la construcción de todos los clasificadores necesarios para llevar a cabo la metodología.

Sin embargo, como se observó en la Sección 6.1, el comportamiento de SVM dentro del método BEM no genera información útil; el valor del área bajo la curva (AUC) incrementa directamente hacia su valor máximo al aumentar el número de variables, lo que significa que el método BEM no sería capaz de reducir el conjunto F, dejando $Q = F$, por lo tanto,

la variación del valor umbral h no tendría ningún sentido en este caso. Esto implica que la metodología no generaliza a todas las técnicas de aprendizaje automático, pero no significa que no deba utilizarse, de hecho, muestra un buen rendimiento para el caso de regresión logística, tanto para seleccionar un buen conjunto de variables como para construir un buen clasificador, tal y como se observa en la Tabla 6.3. La diferencia más significativa entre los valores de AUC para los conjuntos de entrenamiento y de prueba es de aproximadamente $-4\,\%$ (0.934 a 0.896) para el conjunto de datos bancarios, lo cual muestra un buen resultado sobre el clasificador construido. Resultados que se logran con un menor número de variables, desde un $33\,\%$ para el conjunto de datos de ataques cardiacos hasta un máximo de $85\,\%$ para el conjunto de predicción de púlsares, lo cual demuestra ser eficiente al evitar el sobreajuste de los clasificadores.

Otro aspecto importante es el uso del modelo de regresión logística al momento de aplicar el método BEM debido a la complejidad para ser utilizado por las técnicas de SVM y MLP. Recordando que dentro del método de selección híbrido se busca realizar una primera aproximación que sea competitiva respecto a otros métodos de selección de variables para después construir clasificadores más complejos. Este enfoque funciona bien para DT y MLP en los conjuntos estudiados, sin embargo, se observa una mayor diferencia entre los valores de AUC cuando el número de variables eliminadas es menor, especialmente en el conjunto del bosón de Higgs, donde se eliminó aproximadamente la mitad de las variables y el valor de AUC de los conjuntos de entrenamiento y prueba se aproxima mejor en el método de LR. Esto podría indicar que el uso de LR para la metodología de selección de variables es el adecuado, dejando para un paso posterior la construcción de clasificadores más complejos.

La última observación sobre estos conjuntos de datos es el resultado que ofrece SVM. Se tiene un comportamiento monótono creciente del AUC con respecto al incremento del número de variables, con una tendencia a su valor máximo. Esto indica que hacer uso de esta técnica para la selección de variables no es adecuado. Nuevamente, esto no significa que SVM no deba usarse, sino que se debe encontrar la SVM adecuada para el problema,

por ejemplo, utilizando diferentes configuraciones como las mencionadas en la Sección 3.2.3. Como se mencionó anteriormente, la metodología puede dividirse para primero seleccionar el conjunto de variables y luego construir un clasificador diferente. Esto fue lo que se realizó para construir y seleccionar una *buena* SVM para cada uno de los conjuntos, y dio un buen rendimiento, incluso para el conjunto del bosón de Higgs.

7.2. Conjunto de datos de τ

Con los resultados favorables de la metodología (modificada) para la selección de variables en cada uno de los conjuntos de datos independientes, se procede a aplicarla sobre el conjunto de datos de candidatos de τ. Lo primero que se debes destacar es la cardinalidad de su correspondiente conjunto F, que consta de 162 variables. Esta cardinalidad difiere en gran medida de los conjuntos F anteriores. Por esta razón, no fue posible construir una gran cantidad de clasificadores basados en SVM y MLP. La metodología está limitada al número de atributos que definen cada instancia de un conjunto D, al menos para clasificadores complejos. Sin embargo, como ya se ha mencionado, la propuesta es fácilmente modificable.

Específicamente, la capacidad de utilizar solo el conjunto de variables proporcionado por la metodología aplicada al modelo de regresión logística para construir clasificadores de SVM y MLP generó resultados favorables, como se muestra en la Tabla 6.4. A pesar de no haber construido la totalidad de los clasificadores para SVM y MLP, la diferencia entre los valores de AUC para los conjuntos de entrenamiento y prueba no es considerable. Además, al igual que en el caso de los conjuntos independientes, la reducción de variables es significativa.

Ahora bien, estos resultados ayudan a resolver el problema de clasificación binaria desde el punto de vista de ciencias de la información, pero además se necesita dar una interpretación para la física de partículas. Dado que se está trabajando en el experimento de Belle II y se espera que la metodología pueda ser usada para el estudio de cualquier clasificador se proponen dos opciones.

1.- Utilizar el resultado del clasificador como la solución final, es decir, se proporciona la información del posible candidato a un decaimiento específico al clasificador construido y esperar su respuesta, ya sea positiva o negativa y a partir de la clasificación de todo el conjunto de candidatos, podemos observar las distribuciones de las variables relevantes para la física de partículas en cada conjunto. Esto incluiría distribuciones como el momento, masa, pseudorapidez, entre otras. El objetivo sería comparar estas distribuciones de datos clasificados con las distribuciones de los datos generados por el modelo de Monte Carlo (MC) y si hay diferencias significativas entre estas distribuciones, se requeriría investigar la causa de ello, que podrían deberse a la construcción del clasificador o a la medición de los datos experimentales. Analizar y comprender la fuente del error sería una discusión importante en la física de partículas, ya que una mala interpretación podría tener un impacto considerable en las leyes de la física moderna. Esto podría llevar a realizar costosos experimentos para una comparación exhaustiva.

2.- Otorgar menor peso al poder de clasificación de un clasificador específico, es decir, en lugar de tomar simplemente la salida discreta del clasificador (0 o 1), consideraríamos la salida de manera estadística, con valores entre 0 y 1. Cada instancia tendría un valor de probabilidad que representa la confianza del clasificador en su clasificación correcta. Esto nos permitiría establecer una distribución de probabilidad para el clasificador, creando así una nueva variable para el conjunto de datos D y con esto, podríamos mantener los estudios estadísticos en la física de partículas y al mismo tiempo aprovechar la nueva información proporcionada por el clasificador.

Para determinar si una instancia se clasifica como positiva (1) o negativa (0) según estas distribuciones, podríamos establecer un valor umbral. Este proceso es similar al que utilizamos al evaluar las métricas de la Sección 3.3 basadas en la matriz de confusión. La diferencia radica en el orden en el que se aplicarían los cortes en este enfoque, ya que podríamos utilizar diferentes distribuciones para realizar estos cortes, por ejemplo, en las Figuras 7.1 y 7.2 se muestran ejemplos de las distribuciones de probabilidad para las técnicas de regresión logística

(LR) y árboles de decisión (DT) respectivamente, utilizando conjuntos de entrenamiento. Estas distribuciones nos permiten encontrar un valor adecuado para establecer si una instancia es clasificada como positiva o negativa.

En ambos casos, se brinda una manera más flexible y poderosa de utilizar los resultados del clasificador, permitiendo incorporar información estadística adicional a los estudios en la física de partículas y mejorando la eficiencia en el análisis de candidatos de decaimiento.

En cualquiera de los casos se requiere un consenso de la comunidad de física de Belle II, que presenta una oportunidad para futuros trabajos.

7.3. Conclusiones

A través de este trabajo, se ha desarrollado una metodología para la selección de variables y construcción de clasificadores de forma conjunta. Sin embargo, es importante destacar que esta metodología no puede aplicarse directamente a todos los conjuntos de variables que cumplen con las características del conjunto D. Su efectividad dependerá del número de variables, la cardinalidad del conjunto y la complejidad de las técnicas de aprendizaje utilizadas.

Con base en los resultados obtenidos en los diferentes experimentos, se pueden sugerir diferentes enfoques para aplicar la metodología, por ejemplo, para conjuntos de datos D con una cardinalidad de aproximadamente 300,000 unidades y un número máximo de 30 variables ($|F| = 30$), es factible aplicar directamente la metodología utilizando el modelo de regresión logística, para las técnicas de árboles de decisión y perceptrón multicapa, podemos utilizar la metodología modificada que se basa en el modelo de regresión logística, mientras que para máquinas de vectores soporte y perceptrón multicapa, podemos tomar directamente el conjunto de variables seleccionadas por la metodología de regresión logística y construir los clasificadores correspondientes.

Cuando la cardinalidad del conjunto $F \geqslant 100$, la metodología para regresión logística

continúa siendo válida, y lo mismo ocurre con la metodología modificada para árboles de decisión. Sin embargo, para SVM y MLP, será necesario utilizar directamente el conjunto seleccionado por la metodología de regresión logística para construir los clasificadores.

Finalmente, esta metodología como cualquier otra esta sujeta al poder de cómputo disponible, razón por la cual se describió una metodología general, una metodología para la selección de variables y clasificador y algunas modificaciones sobre esta última, resaltando que logramos encontrar un buen clasificador para dar solución a nuestro problema. Sin embargo, no fue posible realizar una comparación directa (datos reales de Belle II) debido a que todavía no existen datos públicos, pero puede ser un buen punto de partida para hacer una transición entre las técnicas de Belle II y el aprendizaje automático para el análisis de decaimientos del leptón tau.

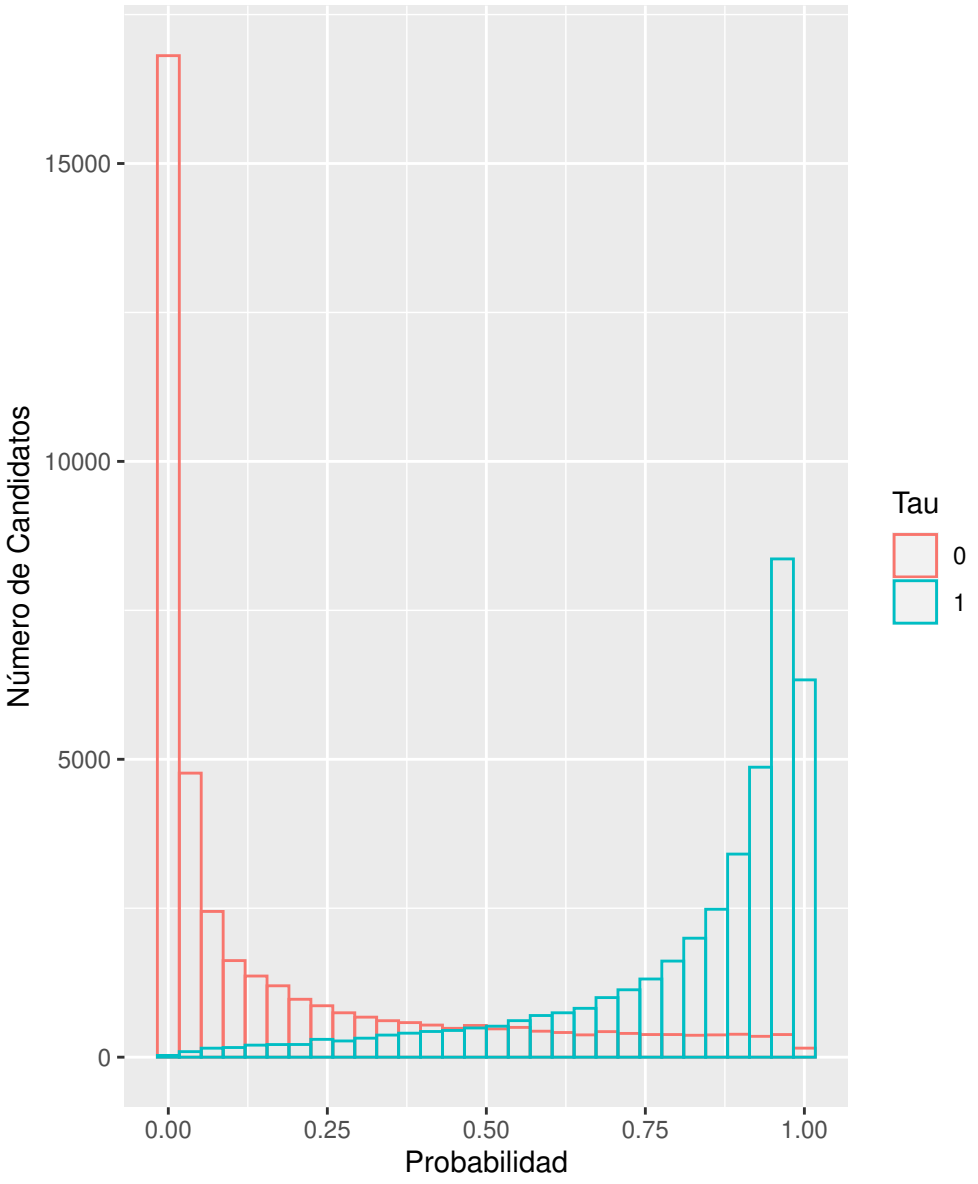

Figura 7.1. Frecuencia de la probabilidad obtenida por la metodología utilizando LR tanto para BEM y el IV.

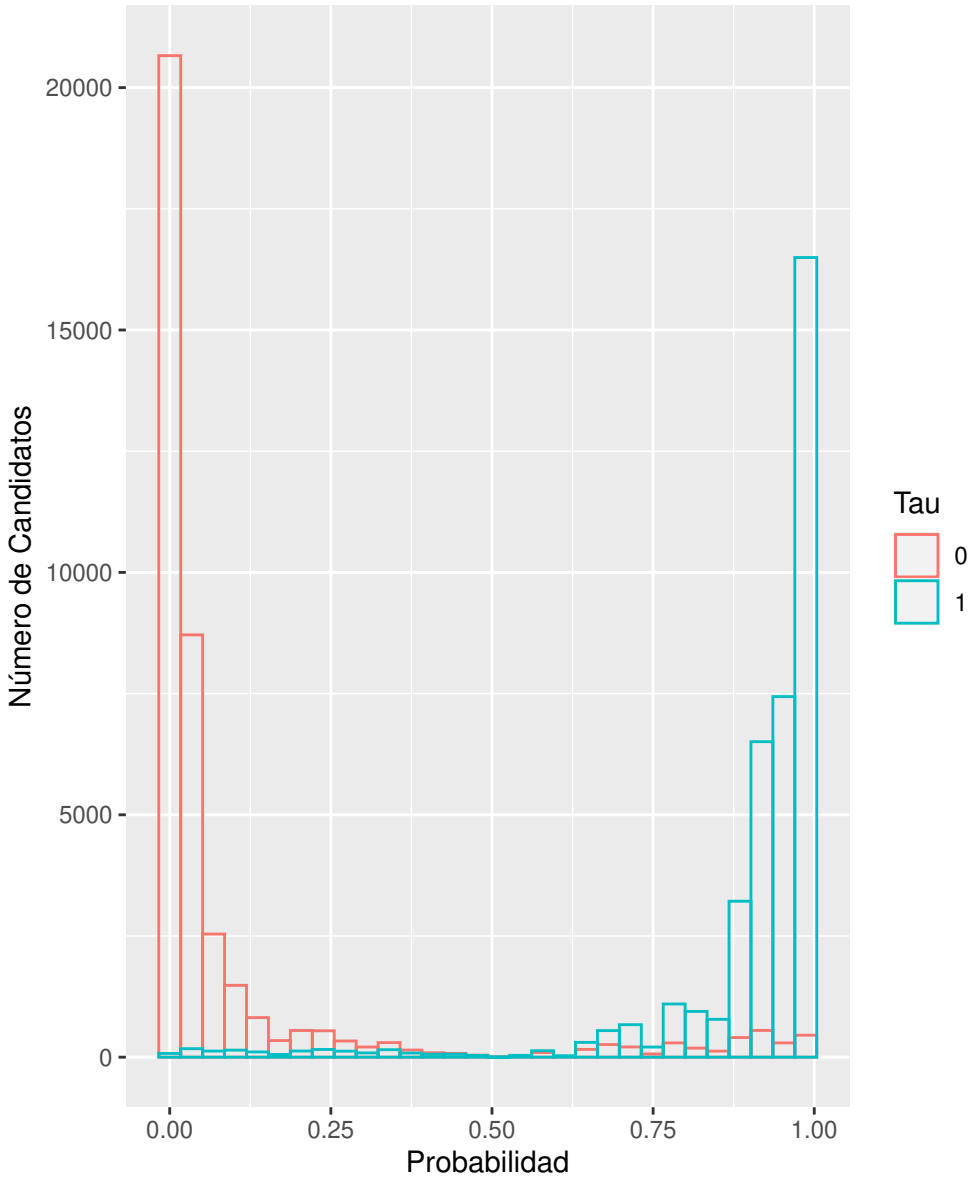

Figura 7.2. Frecuencia de la probabilidad obtenida por la metodología utilizando LR tanto para BEM y DT para el IV.

Bibliografía

[LHC,] The large hadron collider. `https://home.cern/science/accelerators/large-hadron-collider`. Accessed: 2020-01-27.

[Sup,] Superkekb. `http://www-superkekb.kek.jp/`. Accessed: 2020-01-27.

[kag, 2023] (2023). Kaggle: Plataforma de ciencia de datos y competiciones en línea.

[Abudinén and Adachi, 2021] Abudinén, F. and Adachi, I. e. a. (2021). Precise measurement of the D^0 and D^+ lifetimes at belle ii. *Phys. Rev. Lett.*, 127:211801.

[Abudinén and Aggarwal, 2023] Abudinén, F. and Aggarwal, L. e. a. (2023). Measurement of the Λ_c^+ lifetime. *Phys. Rev. Lett.*, 130:071802.

[Adachi and Ahlburg, 2020] Adachi, I. and Ahlburg, P. e. a. (2020). Search for an invisibly decaying $Z^{'}$ boson at belle ii in $e^+e^- \to \mu^+\mu^-(e^\pm\mu^\mp)$ plus missing energy final states. *Phys. Rev. Lett.*, 124:141801.

[Aldrich, 1997] Aldrich, J. (1997). R.A. Fisher and the making of maximum likelihood 1912-1922. *Statistical Science*, 12(3):162 – 176.

[Almasi and Gottlieb, 1989] Almasi, G. S. and Gottlieb, A. (1989). *Highly Parallel Computing*. Benjamin-Cummings Publishing Co., Inc., USA.

[Austin and Tu, 2004] Austin, P. C. and Tu, J. V. (2004). Automated variable selection methods for logistic regression produced unstable models for predicting acute myocardial infarction mortality. *Journal of Clinical Epidemiology*, 57(11):1138–1146.

[Avignone et al., 2008] Avignone, F. T., Elliott, S. R., and Engel, J. (2008). Double beta decay, Majorana neutrinos, and neutrino mass. *Reviews of Modern Physics*, 80(June):481–516.

[Bergmeir and Benítez, 2012] Bergmeir, C. and Benítez, J. (2012). Neural networks in r using the stuttgart neural network simulator: Rsnns. *Journal of Statistical Software*, 46.

[Berners-Lee et al., 2000] Berners-Lee, T., Fischetti, M., and Dertouzos, M. L. (2000). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. HarperInformation.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

[Brown and Davis, 2006] Brown, C. D. and Davis, H. T. (2006). Receiver operating characteristics curves and related decision measures: A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 80(1):24–38.

[Bruce and Bruce, 2017] Bruce, P. and Bruce, A. (2017). *Practical Statistics for Data Scientists: 50 Essential Concepts*. O'Reilly Media.

[Brüning et al., 2004] Brüning, O. S., Collier, P., Lebrun, P., Myers, S., Ostojic, R., Poole, J., and Proudlock, P. (2004). *LHC Design Report*. CERN, Geneva.

[Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

[Cogan et al., 2015] Cogan, J., Kagan, M., Strauss, E., and Schwartzman, A. (2015). Jet-Images: Computer Vision Inspired Techniques for Jet Tagging. *JHEP*, 02:118.

[Commons, 2023] Commons, W. (2023). File:standard model of elementary particles.svg — wikimedia commons, the free media repository. [Online; accessed 8-June-2023].

[Cox, 1958] Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2):215–242.

[Cristianini and Ricci, 2008] Cristianini, N. and Ricci, E. (2008). *Support Vector Machines*, pages 928–932. Springer US, Boston, MA.

[D, 2021] D, P. K. (2021). Higgs bosons and a background process.

[Dal Pozzolo et al., 2014] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., and Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Systems with Applications*, 41:4915–4928.

[de Oliveira et al., 2016] de Oliveira, L., Kagan, M., Mackey, L., Nachman, B., and Schwartzman, A. (2016). Jet-images — deep learning edition. *JHEP*, 07:069.

[Esmail et al., 2019] Esmail, W., Stockmanns, T., and Ritman, J. (2019). Machine Learning for Track Finding at PANDA. In *Connecting the Dots and Workshop on Intelligent Trackers (CTD/WIT 2019) Valencia, Valencia, Spain, April 2-5, 2019*.

[et al. (Particle Data Group), 2014] et al. (Particle Data Group), K. O. (2014). *Chin. Phys. 090001*, C38.

[Fürnkranz,] Fürnkranz, Johannes, e. t. b. y. pages 263–267. Springer US, Boston, MA.

[Genuer et al., 2010] Genuer, R., Poggi, J.-M., and Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern Recognition Letters*, 31(14):2225–2236.

[Gupta et al., 2022] Gupta, R., Bhattacharya, T., and Yoon, B. (2022). Ai and theoretical particle physics.

[Haake and Loizides, 2019] Haake, R. and Loizides, C. (2019). Machine Learning based jet momentum reconstruction in heavy-ion collisions. *Phys. Rev.*, C99(6):064904.

[HEP ML Community,] HEP ML Community. A Living Review of Machine Learning for Particle Physics.

[Hsu et al., 2011] Hsu, H.-H., Hsieh, C.-W., and Lu, M.-D. (2011). Hybrid feature selection by combining filters and wrappers. *Expert Syst. Appl.*, 38:8144–8150.

[Jadach and Ward, 2000] Jadach, S. and Ward, B. (2000). The precision monte carlo event generator kk for two-fermion final states in e+e- collisions. *Computer Physics Communications*, 130:260–325.

[Jain and Singh, 2018] Jain, D. and Singh, V. (2018). An efficient hybrid feature selection model for dimensionality reduction. *Procedia Computer Science*, 132:333–341.

[James et al., 2013] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R.* Springer.

[Keck et al., 2019] Keck, T. et al. (2019). The Full Event Interpretation. *Comput. Softw. Big Sci.*, 3(1):6.

[Kou et al., 2018] Kou, E., Urquijo, P., community, B. T., and Collaboration, B. I. (2018). The belle ii physics book.

[Lehmann, 1993] Lehmann, E. L. (1993). The fisher, neyman-pearson theories of testing hypotheses: One theory or two? *Journal of the American Statistical Association*, 88(424):1242–1249.

[López Castro and Quintero, 2013] López Castro, G. and Quintero, N. (2013). Bounding resonant majorana neutrinos from four-body b and d decays. *Phys. Rev. D*, 87:077901.

[Luchman, 2013] Luchman, J. N. (2013). CHAID: Stata module to conduct chi-square automated interaction detection. Statistical Software Components, Boston College Department of Economics.

[Mandal et al., 2021] Mandal, M., Singh, P., Ijaz, M. F., Shafi, J., and Sarkar, R. (2021). A tri-stage wrapper-filter feature selection framework for disease classification. *Sensors*, 21:5571.

[Mann and Primakoff, 1977] Mann, A. K. and Primakoff, H. (1977). Neutrino oscillations and the number of neutrino types. *Phys. Rev. D*, 15:655–665.

[McCormack and Ganai, 2019] McCormack, P. and Ganai, M. (2019). Identifying Merged Tracks in Dense Environments with Machine Learning. In *Connecting the Dots and Workshop on Intelligent Trackers (CTD/WIT 2019) Valencia, Valencia, Spain, April 2-5, 2019*.

[Mohtashami and Eftekhari, 2018] Mohtashami, M. and Eftekhari, M. (2018). A hybrid filter-based feature selection method via hesitant fuzzy and rough sets concepts. *Iranian journal of fuzzy systems*.

[Moll, 2011] Moll, A. (2011). The software framework of the Belle II experiment. *J. Phys. Conf. Ser.*, 331:032024.

[Novakovic et al., 2011] Novakovic, J., Strbac, P., and Bulatović, D. (2011). Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research*, 21:119–135.

[Powers, 2020] Powers, D. M. W. (2020). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.

[Quinlan, 1986] Quinlan, J. R. (1986). Induction of decision trees. 1(1):81–106.

[Quinlan, 1996] Quinlan, J. R. (1996). Improved use of continuous attributes in c4.5. 4(1):77–90.

[Quintero, 2014] Quintero, N. (2014). *Estudios de violación del número leptónico en procesos resonantes inducidos por un neutrino de mayorana*. PhD thesis.

[Raj, 2020] Raj, P. (2020). Predicting pulsar star.

[Rakotomamonjy, 2003] Rakotomamonjy, A. (2003). Variable selection using svm based criteria. *J. Machine Learning Research: Special Issue on Variable and Feature Selection*, 3:1357 – 1370.

[Rashmi, 2020] Rashmi (2020). Banking dataset classification.

[Russell et al., 2004] Russell, S., Norvig, P., and Rodríguez, J. (2004). *Inteligencia artificial: un enfoque moderno*. Colección de Inteligencia Artificial de Prentice Hall. Pearson Educación.

[Sagawa, 1996] Sagawa, H. (1996). Kekb and belle experiment. *Il Nuovo Cimento A*, 109(6-7):1055–1060.

[Sheta, 2021] Sheta, P. (2021). Heart attack.

[Stock and Stock, 2013] Stock, W. G. and Stock, M. (2013). *Handbook of Information Science*. De Gruyter Saur, Berlin, Boston.

[Tegenfeldt, 2007] Tegenfeldt, F. (2007). TMVA - Toolkit for multivariate data analysis with ROOT. PHYSTAT-LHC 2007.

[Ullman and Widom, 2001] Ullman, J. D. and Widom, J. (2001). *A First Course in Database Systems*. Prentice Hall PTR, USA, 2nd edition.

[Viharos et al., 2021] Viharos, Z., Kis, K., Fodor, , and Büki, M. (2021). Adaptive, hybrid feature selection (ahfs). *Pattern Recognition*, 116:107932.

[Ypma, 1995] Ypma, T. J. (1995). Historical development of the newton–raphson method. *SIAM Review*, 37(4):531–551.

[Yu, 2019] Yu, S. (2019). Electron Neutrino Energy Reconstruction in NOvA Using CNN Particle IDs. In *Meeting of the Division of Particles and Fields of the American Physical Society (DPF2019) Boston, Massachusetts, July 29-August 2, 2019*.

[Zeng, 2013] Zeng, G. (2013). Metric divergence measures and information value in credit scoring. *Journal of Mathematics*, 2013.

[Zou et al., 2007] Zou, K. H., O'Malley, A. J., and Mauri, L. (2007). Receiver-operating characteristic analysis for evaluating diagnostic tests and predictive models. *Circulation*, 115(5):654–657.

Apéndice A

Descripción del conjunto de datos de taus

De manera general, la reconstrucción de los diferentes decaimientos en el detector Belle II se basan en las características de partículas cargadas como el electrón (e^-), el muon (μ), entre otras, pero como se mencionó durante el presente trabajo, dicho detector no es capaz de identificar su naturaleza completamente, por lo que su alcance es final son las trazas. Estas pueden definirse a través del siguiente conjunto de datos (ejemplo de la traza número durante la adquisición de datos).

```
TrackFitResult#0:
  get4Momentum(): (-0.348079,-0.048974,0.364701,0.525399)
  getChargeSign(): 1
  getCotTheta(): 1.03753
  getCov(): size(15)
  getCovariance5():
  getCovariance6():
  getD0(): 0.0118526
  getEnergy(): 0.525399
```

getHitPatternCDC(): 00111111101111111111111111...

getHitPatternVXD(): 00000000010101010101010100000001

getMomentum(): (−0.348079, −0.048974, 0.364701)

getOmega(): 0.0123903

getPValue(): 0.0330003

getParticleType(): <type: pi+>

De manera similar se tiene la descripción para la partículas neutras llamadas fotones.

ECLCluster#1:

getAbsZernike40(): 0.861621

getAbsZernike51(): 0.530859

getClusterHadronIntensity(): 0.00757332

getClusterId(): 1

getClusterPosition(): (106.011,...

getConnectedRegionId(): 2

getCovarianceMatrix3x3():

getDeltaL(): 0

getDeltaTime99(): 13.916

getDetectorRegion(): 2

getE1oE9(): 0.732422

getE9oE21(): 0.999023

getEnergy(16): 0.173816

getEnergy(32): nan

getEnergyHighestCrystal(): 0.127081

getEnergyRaw(): 0.168958

getHypotheses(): 16

getLAT(): 0.0771484

getMaxECellId(): 5744

```
getMinTrkDistance(): 9.52148
getNumberOfCrystals(): 2.92969
getNumberOfHadronDigits(): 0.997066
getPhi(): −0.718095
getPulseShapeDiscriminationMVA(): 0.000751495
getR(): 141.842
getRelationsWith(KlIds): size(0)
getSecondMoment(): 0.625
getStatus(): 4
getTheta(): 1.69361
getTime(): −5.37109
getUncertaintyEnergy(): 0.00498047
getUncertaintyPhi(): 0.00839844
getUncertaintyTheta(): 0.00664062
getUniqueId(): 205001
getZernikeMVA(): 0.0595703
hasFailedFitTime(): False
hasFailedTimeResolution(): False
hasHypothesis(16): True
hasHypothesis(32): False
hasPulseShapeDiscrimination(): True
hasTriggerClusterMatching(): False
isNeutral(): True
isTrack(): False
isTriggerCluster(): False
```

Finalmente, cada una de las partículas necesarias para realizar el análisis de un decaimiento podrá usar la información correspondiente a su tipo.

Apéndice B

Selección de variables

El método híbrido de selección de variables descrito en la sección 3.1.4 nos permite disminuir de manera considerable el número de variables a ser usadas en diferentes clasificadores. Dicho método considera calcular el valor de la información para creación de lista ordenada de variables, lo cual se debe a que fue el mejor método para crear la lista. A continuación se describen los resultados obtenidos utilizando diferentes métodos de ordenamiento para un conjunto de prueba de pares de taus.

B.0.1. Métodos de ordenamiento

A continuación se presentan tres métodos de "filtrado" los cuales serán evaluados al entrenar los modelos de clasificación. Para comparar los diferentes métodos de filtrado se calculó el AUC de la curva ROC. Recordando que para la selección de variables solo utilizaremos el modelo de regresión logística, se aplicaron diferentes valores de umbral para el caso del cálculo de la varianza. Para los casos del calculo de la separación y valor de la información, se opto por generar subconjuntos de variables agregando una a una de las variables según el valor obtenido.

Varianza

El estudio de un "buen" conjunto de variables para el entrenamiento de los diferentes modelos se inició con el cálculo de la varianza de cada una de las variables utilizando el conjunto total de ejemplos. De acuerdo a la sección 3.1.1, TMVA nos permite imponer un límite inferior para la varianza pero no nos permite observar de manera directa las variables que cumplen con cierto límite, entonces se probaron diferentes límites de varianza para observar el comportamiento de la gráfica ROC y su correspondiente AUC al entrenar el modelo de regresión logística. En la figura B.1 se realizaron los filtros en la varianza correspondiente a 1, 2, 3 y 10, donde no podemos encontrar un cambio significativo en el valor de AUC de cada uno de los modelos. Lo cual no permite diferenciar entre diferentes conjuntos de variables y razón por la cual no se realizaron pruebras con otros modelos de clasificación.

Separación

Para la revisión de este método de selección primero se calculó la separación y se obtuvo un ranqueo de las variables para posteriormente entrenar el modelo de regresión logística con diferentes conjuntos de variables, los cuales están compuestos con las n variables con el mejor ranqueo. En las figuras B.2, B.3, B.4, B.5 y B.6 se puede observar el comportamiento del modelo con 15, 18, 25, 30 y 40 variables usando el ordenamiento por separación. En donde podemos observar que existe un incremento significativo en el AUC respecto al aumento de variables. A pesar de observar un incremento significativo es necesario realizar un estudio más profundo para determinar el número de variables a seleccionar.

Valor de la información

Para observar un mejor comportamiento del rendimiento de los modelos de clasificación respecto al número de variables se graficó el AUC obtenida por cada modelo al incrementar el número de variables siguiendo el ranqueo obtenido al calcular el valor de la información. En la figura B.7, se puede observar un incremento gradual del AUC al incrementar el número de

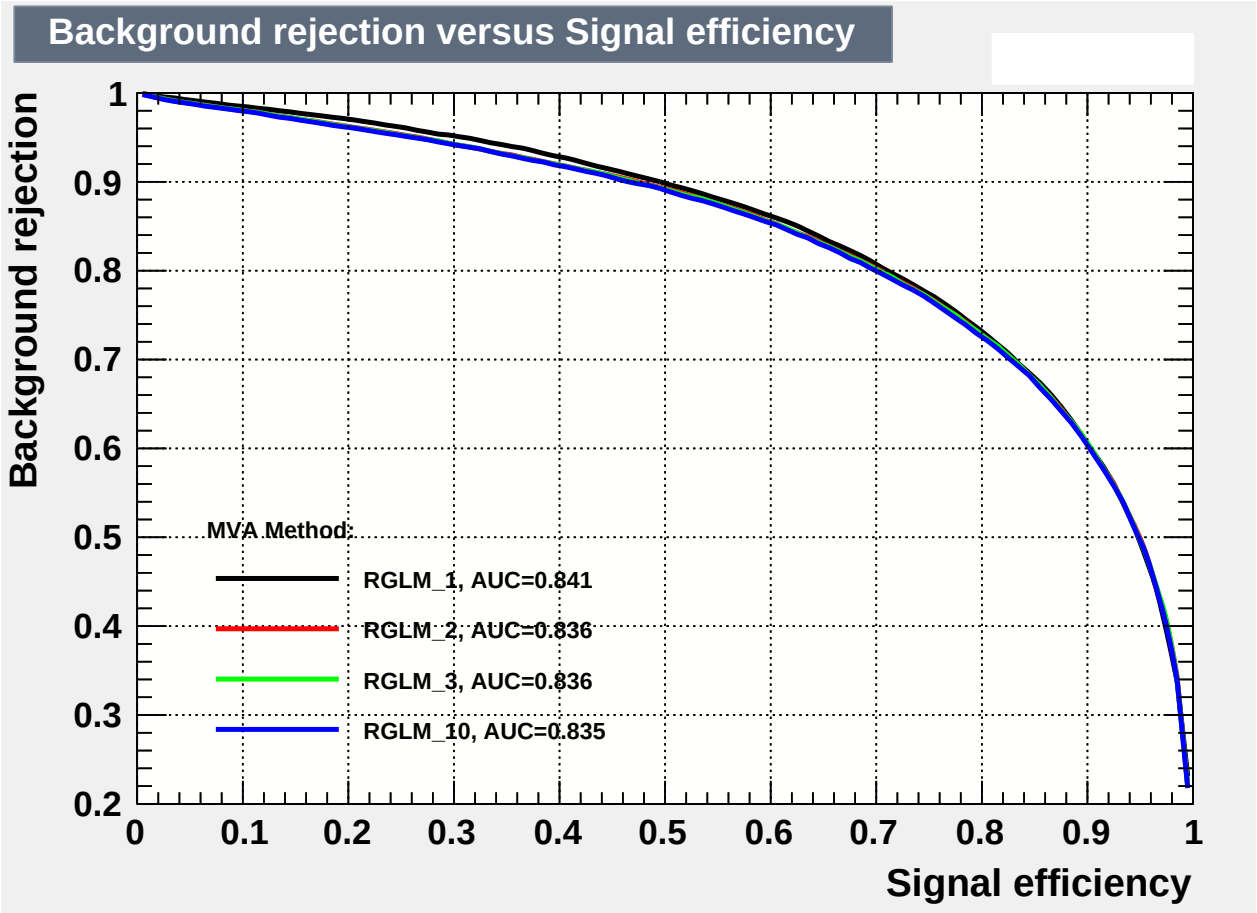

Figura B.1. Rendimiento del modelo de regresión logística utilizando el filtrado por varianza.

variables pero también se observan secciones en las cuales el aumento no es significativo por lo que podrían omitirse dichas variables.

B.0.2. Métodos de envoltura

Los resultados para los métodos de eliminación hacia atrás y selección hacia adelante paso a paso aplicados al modelo de regresión logística fueron similares. Por lo que tomamos como referencia el método de eliminación hacia atrás y observamos el valor del AUC de la curva ROC para el subconjunto de variables obtenido.

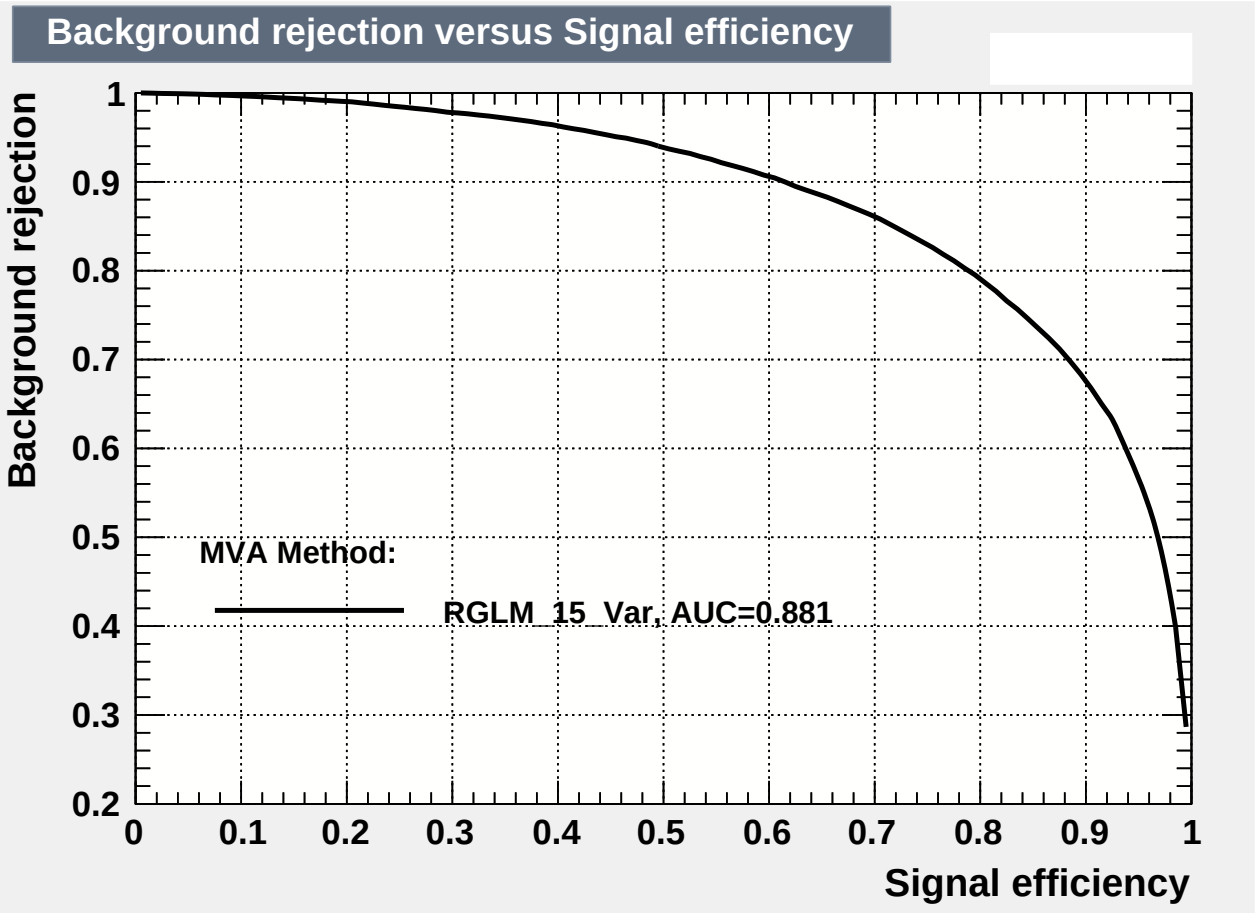

Figura B.2. Rendimiento del modelo de regresión logística utilizando el filtrado por separación utilizando 15 variables.

Eliminación hacia atrás y selección hacia adelante

Los métodos de eliminación hacia atrás y de selección hacia adelante son métodos que realizan una búsqueda semi exhaustiva del conjunto de variables óptimo, es decir, no buscan entre los 2^n subconjuntos de variables posibles, donde n es el número total de variables, pero puede llegar a realizar búsquedas de varios ordenes de magnitud. El resultado de los métodos de eliminación hacia atrás y selección hacia adelante aplicados al modelo de regresión logística fue similar, la única diferencia fue el tiempo de computo utilizado. Ambos métodos encontraron un conjunto de 114 variables de un total de 157, pero utilizaron 30 y 12 horas de cómputo respectivamente. En la Figura B.8 muestra que el resultado del AUC es similar al de la Figura B.6, pero con un menor número de variables.

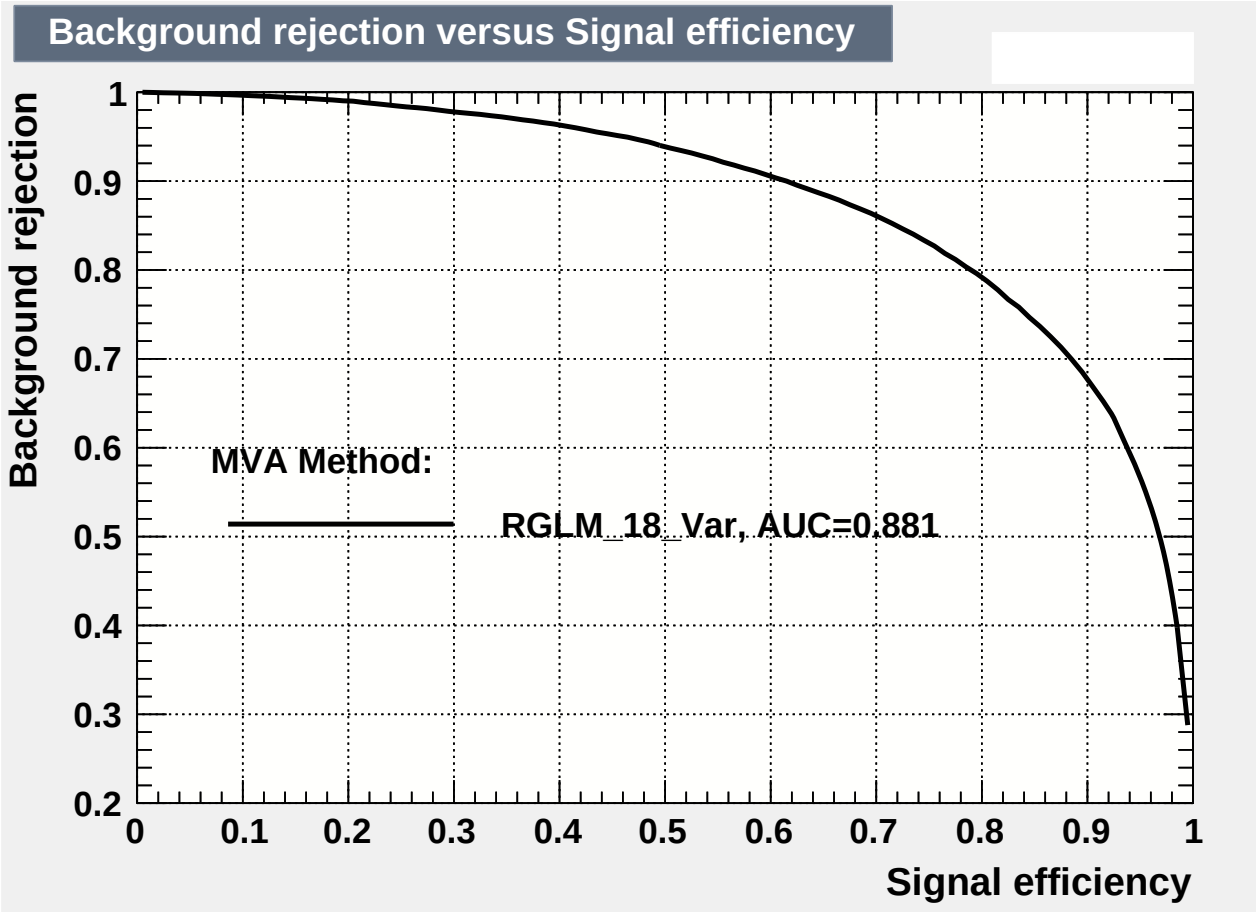

Figura B.3. Rendimiento del modelo de regresión logística utilizando el filtrado por separación utilizando 18 variables.

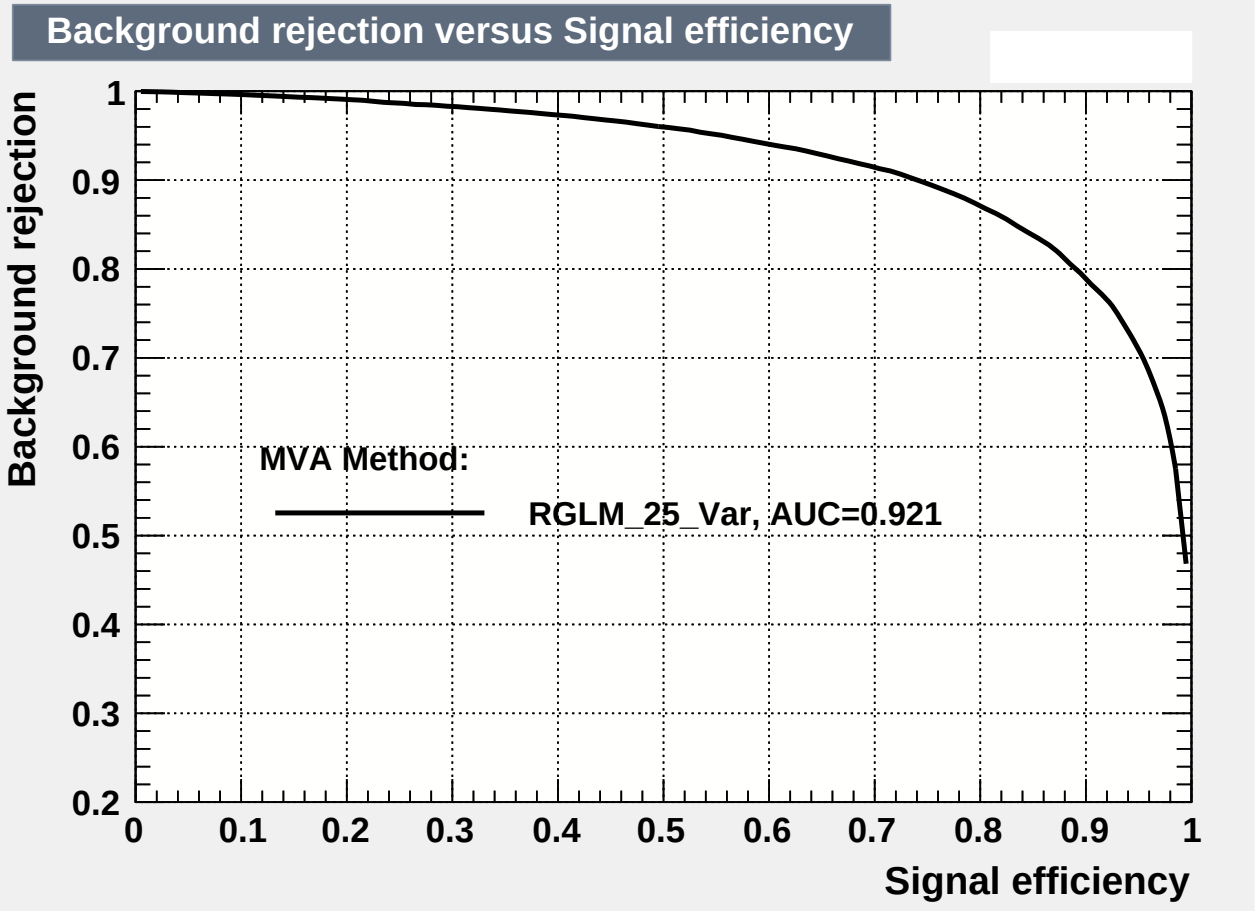

Figura B.4. Rendimiento del modelo de regresión logística utilizando el filtrado por separación utilizando 25 variables.

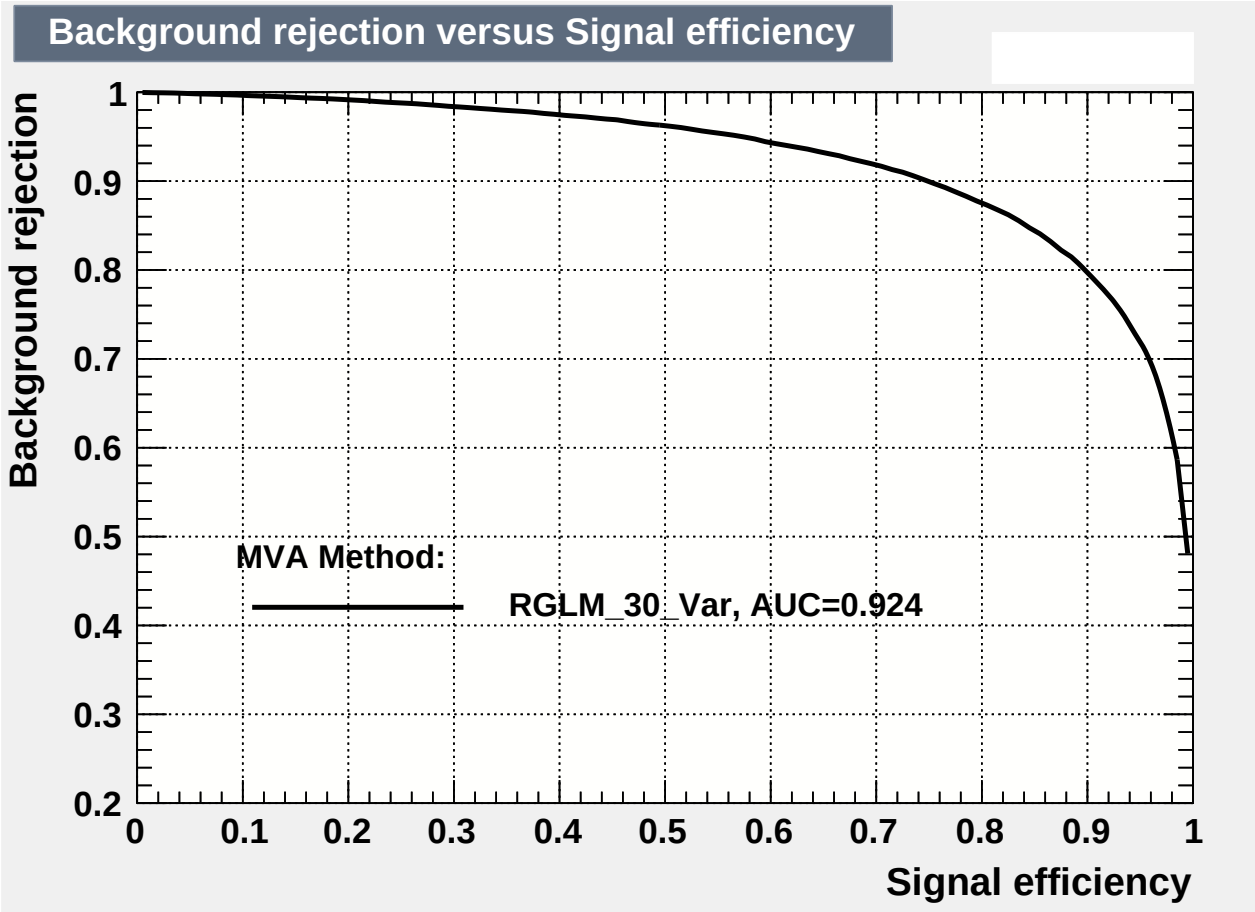

Figura B.5. Rendimiento del modelo de regresión logística utilizando el filtrado por separación utilizando 30 variables.

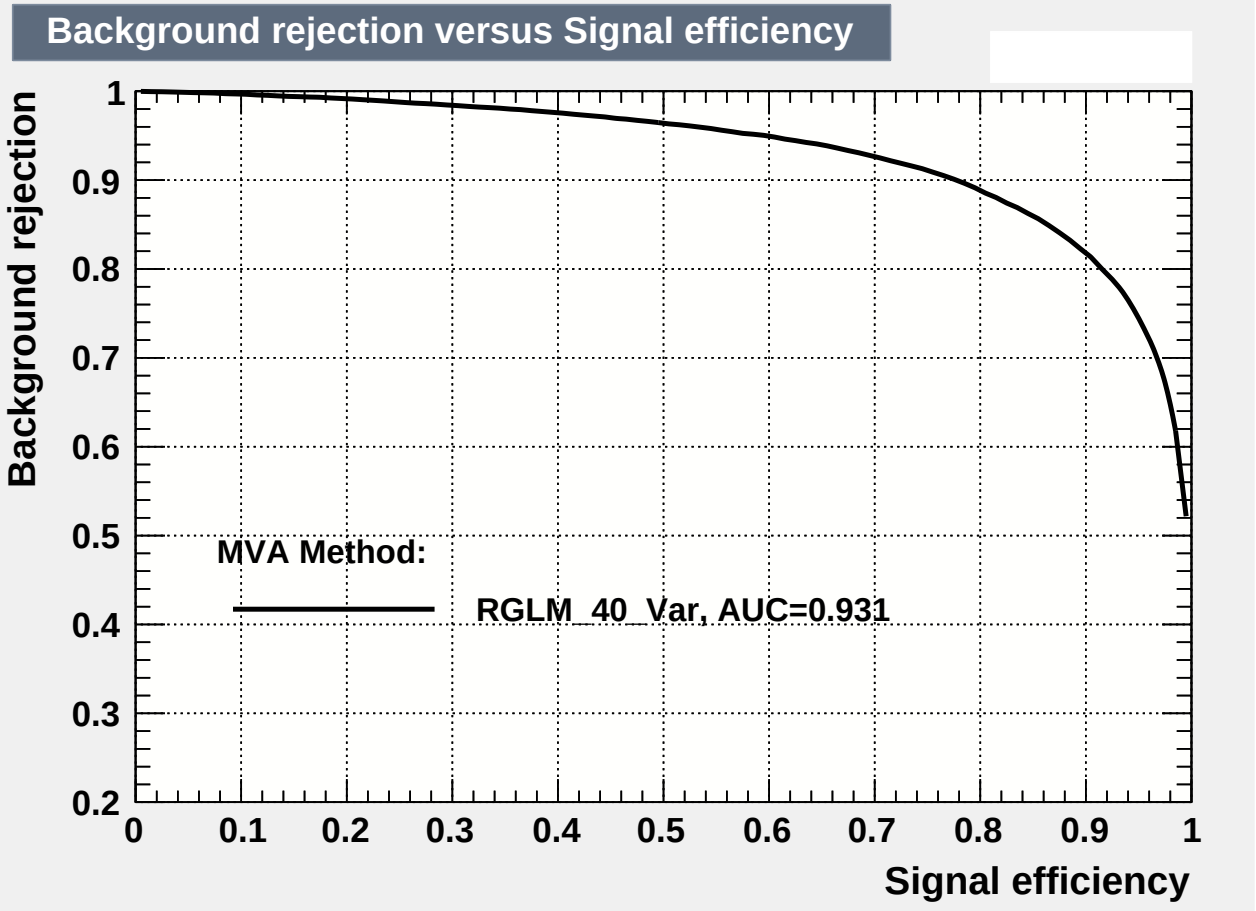

Figura B.6. Rendimiento del modelo de regresión logística utilizando el filtrado por separación utilizando 40 variables.

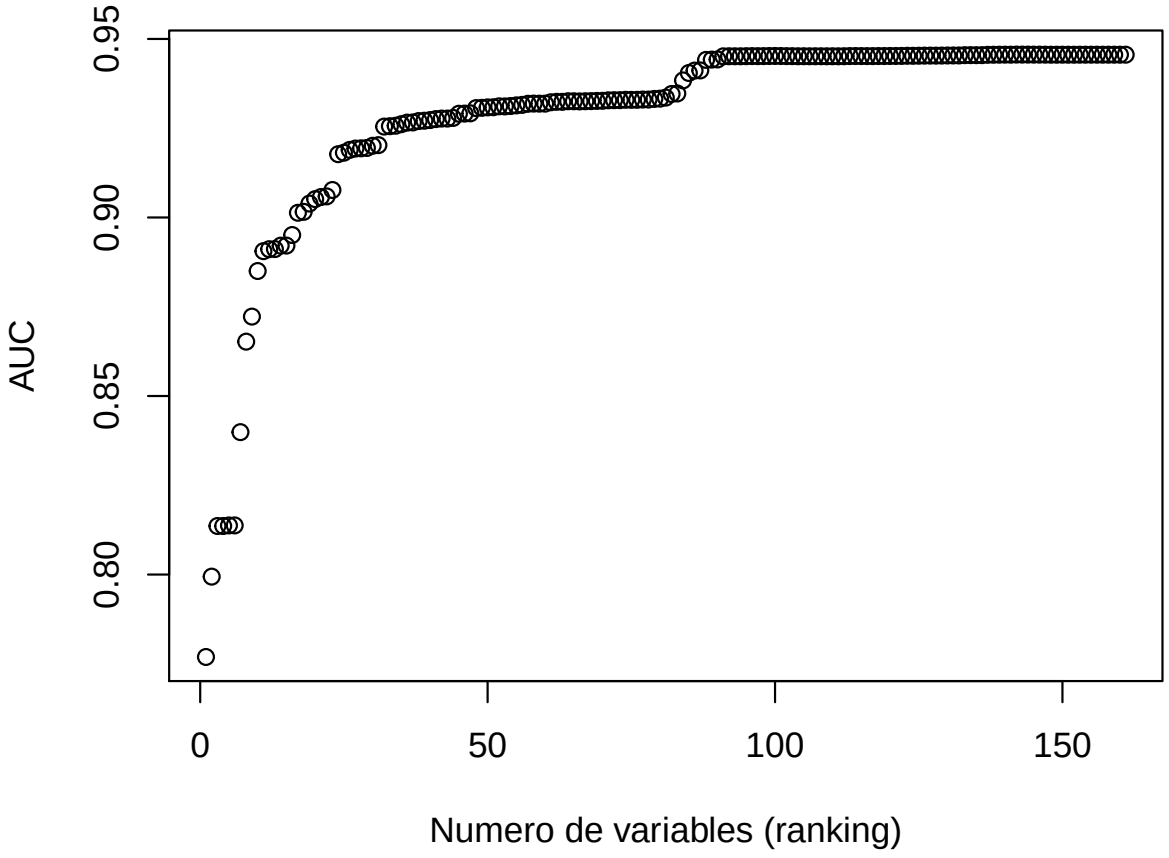

Figura B.7. Rendimiento del modelo de regresión logística incrementando el numero de variables de acuerdo al ranque por el valor de la información.

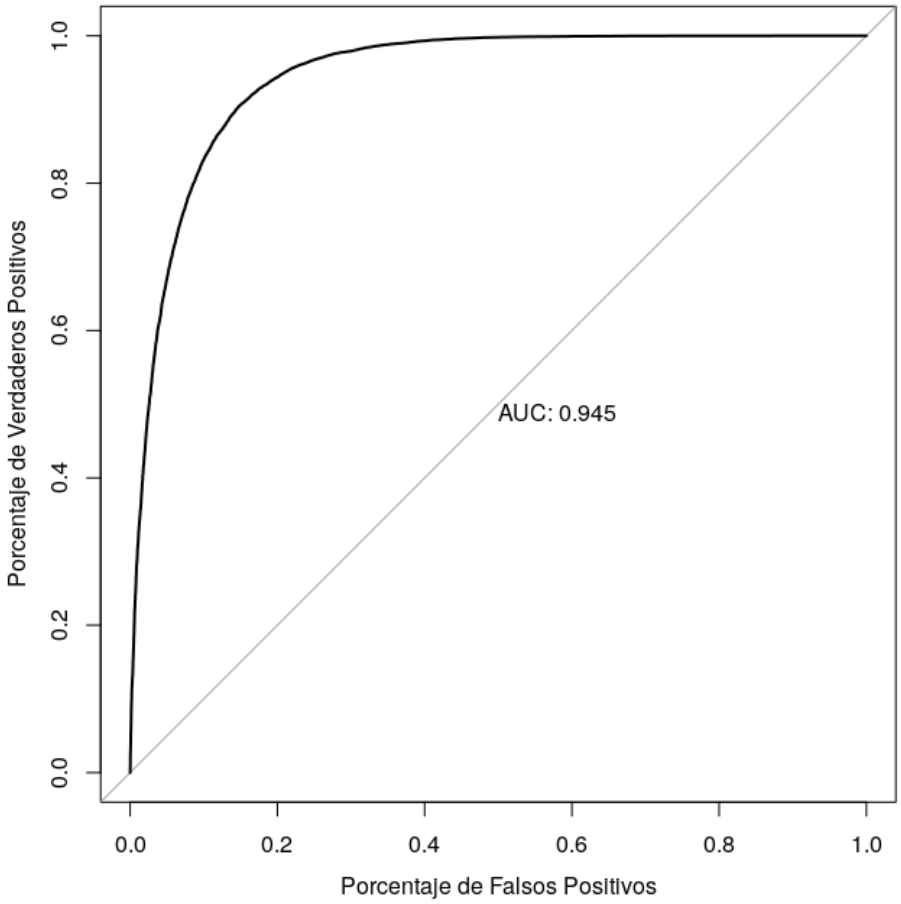

Figura B.8. Rendimiento del modelo de regresión logística aplicando el método de selección hacia adelante y eliminación hacia atrás.