EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación y Desarrollo Tecnológico

Tesis de Maestría

Optimización de Hiperparámetros en Modelos de Aprendizaje Automático: Aplicación en la Predicción de Enfermedades Cardiovasculares

presentada por

L.C.C. Eduardo Sánchez Jiménez

como requisito para obtener el grado de

Maestro en Ciencias de la Computación

Directora de tesis

Dra. María Yasmín Hernández Pérez

Codirector de tesis

Dr. Javier Ortiz Hernández

Cuernavaca, Morelos, México. Enero de 2024.

cenidet
Centro Nacional de Investigación y Desarrollo Tecnológico

EDUCACIÓN
SEP TecNM CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO
TECNOLOGICO NACIONAL DE MEXICO

RECIBIDO
14 DIC 2023
SUBDIRECCIÓN ACADÉMICA

Centro Nacional de Investigación
y Desarrollo Tecnológico
Departamento de Ciencias Computacionales

Cuernavaca, Mor., 13/diciembre/2023

OFICIO No. DCC/209/2023
Asunto: Aceptación de documento de tesis
CENIDET-AC-004-M14-OFICIO

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial de EDUARDO SÁNCHEZ JIMÉNEZ con número de control M22CE005, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis de grado titulado "OPTIMIZACIÓN DE HIPERPARÁMETROS EN MODELOS DE APRENDIZAJE AUTOMÁTICO: APLICACIÓN EN LA PREDICCIÓN DE ENFERMEDADES CARDIOVASCULARES" y hemos encontrado que se han atendido todas las observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

MARÍA YASMÍN HERNÁNDEZ PÉREZ
Directora de tesis

JAVIER ORTIZ HERNÁNDEZ
Codirector de Tesis

ALICIA MARTÍNEZ REBOLLAR
Revisor 1

HUGO ESTRADA ESQUIVEL
Revisor 2

C.c.p. Depto. Servicios Escolares,
Expediente / Estudiante

EDUCACIÓN
14 DIC 2023
CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO
SERVICIOS ESCOLARES
RECIBIDO

Cuernavaca, Mor., 15/diciembre/2023
No. De Oficio: SAC/209/2023
Asunto: Autorización de impresión de tesis

EDUARDO SÁNCHEZ JIMÉNEZ
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS DE LA COMPUTACIÓN
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **"OPTIMIZACIÓN DE HIPERPARÁMETROS EN MODELOS DE APRENDIZAJE AUTOMÁTICO: APLICACIÓN EN LA PREDICCIÓN DE ENFERMEDADES CARDIOVASCULARES"**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"Conocimiento y tecnología al servicio de México"

EDUCACIÓN | TECNOLÓGICO NACIONAL DE MÉXICO
CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO
DIRECCIÓN

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

C. c. p. Departamento de Ciencias Computacionales
Departamento de Servicios Escolares

CMAZ/lmz

A mis abuelos Ernestina† y Juan†
mis padres Erutila y Jeremías
y mi hermana Paula.

Agradecimientos

Al Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCyT) por el programa Sistema Nacional de Posgrados (SNP) por medio del cual me apoyó económicamente como estudiante de tiempo completo.

A mi directora de tesis, la Dra. María Yasmín Hernández Pérez, quien me enseñó sobre la emocionante área del Aprendizaje Automático y quien siempre apoyó mis iniciativas sobre el desarrollo de esta tesis.

Al Dr. Javier Ortiz Hernández por cada una de sus observaciones, sabios consejos y sugerencias a lo largo de esta tesis.

A mi comité tutorial conformado por la Dra. Alicia Martínez Rebollar y el Dr. Hugo Estrada Esquivel, quienes dedicaron parte de su tiempo a las revisiones de este trabajo y ayudaron a a la mejora de este.

Al Tecnológico Nacional de México campus Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), por brindarme la oportunidad de pertenecer a su comunidad estudiantil.

A mis padres, hermanos, sobrinos, cuñada y al resto de mi familia, gracias por su apoyo incondicional y alentarme a seguir adelante.

A mis amig@s Ale, Betty, Dulce, Arely, Eydi, Omar, Miguel, Enrique, Jonathan, Berny, Alejandra, Julia, Antonio, René y al Dr. Alberto, por echarme porras. De igual forma, a los integrantes de **Elba´s Bar**: Isay, Uriel, Javier, Sergio, Toby†, Óscar, Inri e Irvin.

A los profesores y personal del CENIDET, en particular al Dr. José Luis, Dra. Andrea, Dr. Máximo y Dr. Noé, Dr. Joaquín, Dr. Nimrod, Mtra. Cecilia y Lic. Irma.

Resumen

Los modelos de predicción son herramientas analíticas que utilizan datos históricos y estadísticas para realizar inferencias que apoyan en la toma de decisiones, en particular, en el campo médico. Estos modelos están conformados por variables llamadas hiperparámetros que se configuran antes del proceso de entrenamiento, con el objetivo de capturar patrones complejos de los datos. Además, su configuración tiene un impacto directo en el desempeño predictivo. La predicción de enfermedades cardiovasculares se ha convertido en un área de gran interés debido a su impacto en la salud pública. La identificación temprana y precisa de los factores de riesgo y la predicción de eventos cardiovasculares son esenciales para mejorar el diagnóstico y el tratamiento de los pacientes.

En este contexto, este trabajo aborda la configuración de hiperparámetros para maximizar el desempeño de los modelos de predicción de enfermedades cardiovasculares. La propuesta de solución consiste en la aplicación de seis enfoques de optimización con el objetivo de semi-automatizar la búsqueda de los valores para los hiperparámetros de los modelos *Random Forest, Support Vector Machine* y *XGBoost*.

Los resultados de la investigación muestran que enfoques de optimización *Bayesian Optimization, Particle Swarm Optimization*, y *Genetic Algorithm* proporcionaron mejoras significativas en las métricas: *accuracy, recall* y *sensitivity*. En el proceso de optimización del modelo *Random Forest* se identificó que la mejora en el rendimiento está directamente relacionada con el aumento tanto del número y la profundidad de árboles de decisión. El modelo *Support Vector Machine* destaca su desempeño en las métricas *accuracy* y *specificity*, lo que demuestra su fiabilidad a la hora de clasificar correctamente tanto instancias positivas como negativas. Por otro lado, el modelo XGBoost destaca en la métrica de *recall*, es decir, mayor habilidad en la identificación precisa de instancias positivas.

En esta tesis se estudiaron los mecanismos de optimización y se determinaron las configuraciones que mejoran la predicción de enfermedades cardiovasculares. Estos resultados tienen implicaciones importantes en el área de la inteligencia artificial aplicada al dominio médico, y ofrecen una base sólida para investigaciones futuras.

Abstract

Predictive models are analytical tools that use historical data and statistics to make inferences that support decision making, particularly in the medical field. These models consist of variables known as hyperparameters that are configured prior to the training process with the goal of capturing complex data patterns. In addition, their configuration has a direct impact on predictive performance. Cardiovascular disease prediction has emerged as a highly significant area due to its impact on public health. Early and accurate identification of risk factors and prediction of cardiovascular events are essential to improve patient diagnosis and treatment.

In this context, this work addresses the configuration of hyperparameters to maximize the performance of cardiovascular disease prediction models. The proposed solution consists in the application of six optimization approaches with the aim of semi-automating the search for the values of the hyperparameters of the Random Forest, Support Vector Machine and XGBoost models.

The research results show that Bayesian Optimization, Particle Swarm Optimization, and Genetic Algorithm optimization approaches provided significant improvements in the metrics: accuracy, recall, and sensitivity. For the Random Forest optimization, the performance improvement was found to be directly related to increasing the number and depth of decision trees. The Support Vector Machine model stands out for its performance in the accuracy and specificity metrics, demonstrating its reliability in correctly classifying both positive and negative instances. On the other hand, the XGBoost model stands out in the recall metric, i.e. a greater ability to accurately identify positive instances.

In this thesis, we studied the optimization mechanisms and determined the configurations that improve the prediction of cardiovascular disease. These results have important implications in the field of artificial intelligence applied to the medical domain and provide a solid foundation for future research.

Tabla de contenido

Lista de figuras

Página

Lista de tablas

Página

Capítulo 1

Introducción

La predicción enfermedades cardiovasculares mediante modelos de aprendizaje automático es un desafío complejo debido a la interacción de múltiples factores de riesgo, la variabilidad en los datos clínicos y la arquitectura que se define en los modelos. En ese sentido, este capítulo introduce el problema de la configuración eficiente de los hiperparámetros, justifica la relevancia de la investigación, establece el objetivo general de mejorar la precisión de la predicción y define los objetivos específicos, que incluyen la identificación de factores de riesgo relevantes, la selección de modelos y la optimización de hiperparámetros mediante enfoques de optimización. Además, se describe la metodología de solución que se utilizará para lograr estos objetivos, incluyendo la recopilación de datos, el preprocesamiento, la selección de algoritmos y la evaluación del rendimiento.

1.1. Planteamiento del problema

La construcción de modelos predictivos involucra tomar decisiones importantes. Por ejemplo, seleccionar un tipo de enfoque de aprendizaje o el algoritmo adecuado para el conjunto de datos y el tipo de predicción que se desea realizar. Dentro de la selección del algoritmo y su configuración correspondiente, existen un conjunto de variables que se deben ajustar antes de la fase de entrenamiento, llamadas hiperparámetros. El ajuste adecuado de estas variables proporciona al modelo un mejor rendimiento predictivo en la fase de prueba.

Comúnmente, cuando se construye un modelo de aprendizaje automático los hiperparámetros se utilizan en sus valores establecidos de forma predeterminada, que son generalmente propuestos por los desarrolladores de las bibliotecas de aprendizaje automático. Sin embargo, estos valores predeterminados pueden no ser los más adecuados para conjuntos de datos de un dominio en específico. Por otro lado, aplicar un ajuste manual de los hiperparámetros puede resultar en un proceso largo y tedioso, ya que implica probar diferentes combinaciones de valores y evaluar el rendimiento del modelo para cada configuración. Sin embargo, es subjetiva y no garantiza encontrar la mejor configuración.

Al abordar la configuración de hiperparámetros como un problema de optimización, se aprovechan los beneficios de los enfoques exactos y aproximados que buscan encontrar la combinación más efectiva de valores para maximizar la capacidad predictiva del modelo.

1.2. Justificación

La configuración de los hiperparámetros en los algoritmos de aprendizaje automático se ha convertido en una etapa crucial en el desarrollo de modelos de predicción debido a su impacto directo con el rendimiento, es decir, ayudan a determinan cómo se realizará el aprendizaje y la inferencia del modelo. (Zheng et al., 2019). Dado que existen múltiples combinaciones posibles de los hiperparámetros de un modelo, es necesario contar con un enfoque que permita determinar aquella configuración para maximizar el rendimiento del modelo en función de los datos disponibles para su entrenamiento. En consecuencia, se puede concebir al proceso de configuración de hiperparámetros como un problema de optimización, lo que implica la necesidad de

aplicar enfoques que faciliten la configuración semiautomática de los modelos predictivos.

Al considerar mecanismos de optimización, se pueden obtener modelos de predicción de enfermedades cardiovasculares con un rendimiento mejorado en comparación con los modelos entrenados con los valores predeterminados de los hiperparámetros. Esto es especialmente importante cuando se trata de conjuntos de datos grandes y complejos, donde la elección adecuada de los hiperparámetros puede marcar la diferencia entre un modelo con mal desempeño y uno altamente preciso y generalizable (Kotthoff et al., 2017).

1.3. Objetivos

Objetivo general

Determinar la configuración de los hiperparámetros en modelos de aprendizaje automático para maximizar la predicción de enfermedades cardiovasculares mediante enfoques de optimización exactos y aproximados.

Objetivos específicos

1. Analizar los hiperparámetros clave de los algoritmos de aprendizaje automático para conocer su influencia en el rendimiento general de estos.

2. Implementar técnicas de preprocesamiento de datos para maximizar el rendimiento en la fase de entrenamiento de los modelos de aprendizaje automático.

3. Aplicar enfoques de optimización de hiperparámetros que maximicen el desempeño de los modelos de predicción de enfermedades cardiovasculares.

1.4. Metodología de solución

Para abordar el problema de esta investigación, se proposieron distintas actividades en la metodología de solución, dichas actividades son agrupadas y presentadas en La Figura 1.1. En

ella, se describe las actividades que conforman el proceso de optimización de los hiperparámetros y la evaluación de los modelos.

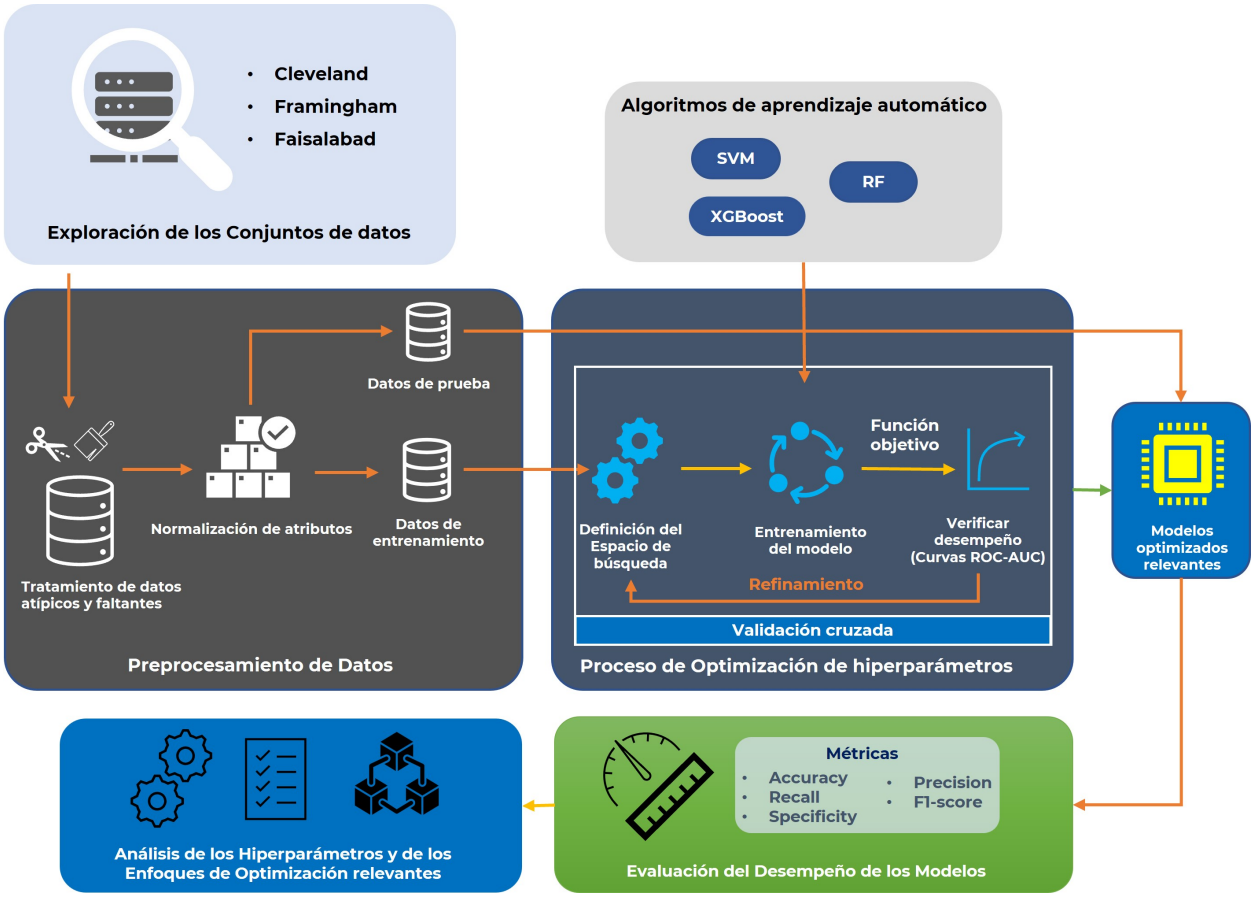


Figura 1.1: Diagrama de bloques de la metodología propuesta.

La actividad inicial fue el análisis exploratorio de los conjuntos de datos Cleveland, Framingham y Faisalabad. Luego, se llevó a cabo la etapa de preprocesamiento de datos para darle formato a la información presente en cada conjunto de datos. Posteriormente, se realizó el proceso de optimización de hiperparámetros, aquí se configuró el espacio de búsqueda y se definió una función para guiar el proceso de optimización (en nuestro caso la métrica ROC-AUC). En la evaluación de los modelos se consideró el análisis de la matriz de confusión y la apicación de las métricas adecuadas al problema. Finalmente, se realizó un análisis de las configuraciones de hiperparámetros relevantes por cada conjunto de datos.

El Capítulo 4 detalla las actividades contempladas en la metodología de solución. Por su parte, el Capítulo 5 muestra los resultados de las actividades y la discusión del proceso experi-

mental.

1.5. Organización de la tesis

El capítulo 2 proporciona la base conceptual para comprender la investigación. Incluye una descripción de los algoritmos y sus hiperparámetros. Se presentan las métricas de evaluación en modelos de clasificación. Posteriormente, se aborda la configuración de hiperparámetros como problema de optimización, donde se profundiza la formalización del proceso de optimización y las fases involucradas. Por último, se describe el funcionamiento de los enfoques de optimización. El capítulo 3 presenta una revisión de los trabajos reportados en la literatura. Se describen los trabajos sobre optimización en el contexto del aprendizaje automático; configuración predeterminada de los hiperparámetros; configuración de hiperparámetros basada en la experiencia del usuario y configuración mediante enfoques de optimización. A continuación, el capítulo 4 describe el análisis exploratorio de datos y el preprocesamiento realizado antes de la optimización de hiperparámetros. Se detallan los procesos de optimización de hiperparámetros, incluyendo la configuración experimental, la inicialización de los parámetros de las metaheurísticas, la definición del espacio de búsqueda, etc. Por su parte, el capítulo 5 muestra los resultados del proceso experimental y se discute el desempeño de cada modelo y la configuración de hiperparámetros relevantes. Además, se contrastan los modelos derivados a partir de los enfoques de optimización y, posteriormente, se muestra un estudio comparativo con el estado del arte. Finalmente, el capítulo 6 da las conclusiones de la investigación y se sugieren áreas para investigaciones futuras y posibles extensiones del trabajo de investigación.

Capítulo 2

Marco teórico

El presente capítulo constituye un pilar esencial en la investigación enfocada en la optimización de hiperparámetros de modelos de aprendizaje automático. En este apartado, se abordan los fundamentos teóricos que sustentan esta investigación, se evalúan de manera precisa las estrategias y enfoques de optimización que guían la construcción de modelos de aprendizaje automático en este ámbito específico. Además, se exploran los tipos de enfermedades cardiovasculares y los patrones que ayudan al diagnóstico

2.1. Aprendizaje automático

El aprendizaje automático se ocupa de desarrollar algoritmos con la capacidad de aprender a partir del histórico de los datos, y constituye, junto con la estadística, la herramienta principal para el análisis inteligente de los datos. Los algoritmos de aprendizaje automático son capaces de analizar grandes cantidades de datos de diversos ámbitos, por ejemplo, el sector salud. Es un sustituto del método convencional de modelización de predicciones que utiliza un ordenador para adquirir conocimientos sobre interacciones complicadas y no lineales entre muchas variables, minimizando la diferencia entre los resultados previstos y los reales (Weng et al., 2017). Dentro del aprendizaje automático, existen vertientes tales como: aprendizaje supervisado, aprendizaje no supervisado, aprendizaje por refuerzo y aprendizaje profundo.

Esta investigación entra en la primera vertiente, un enfoque supervisado se define por el uso de conjuntos de datos etiquetados, diseñados para entrenar o "supervisar" algoritmos para

clasificar datos o predecir resultados con precisión. Usando entradas y salidas etiquetadas, el modelo puede medir su precisión y aprender con el tiempo. Los tipos de problemas que se pueden abordar con aprendizaje supervisado son: clasificación y regresión. La clasificación utiliza un algoritmo para asignar con exactitud los datos de prueba en categorías específicas. Además, reconoce entidades específicas dentro del conjunto de datos e intenta sacar algunas conclusiones sobre cómo se deben etiquetar o definir esas entidades. Por otro lado, la regresión se emplea para comprender la relación entre las variables dependientes e independientes. Es frecuentemente utilizada con el propósito de realizar estimaciones futuras, como pronosticar los ingresos por ventas de una empresa específica ((Julianna y IBM, 2021).

Los algoritmos para trabajar con aprendizaje automático son *Logistic Regression*, *Decision Trees*, *Random Forest*, *Support Vector Machines*, *K-Nearest Neighbors*, *Extreme Gradient Boosting*, *Naive Bayes*. En esta investigación, se han tomado en consideración los algoritmos RF, SVM y XGBoost, ya que, según los informes de la literatura, han demostrado un rendimiento sobresaliente en la tarea de predicción de enfermedades cardiovasculares. A continuación, se presentan descripciones detalladas de estos algoritmos de aprendizaje automático.

2.1.1. Hiperparámetros vs parámetros

Los hiperparámetros y los parámetros son componentes fundamentales en los algoritmos de aprendizaje automático, y tienen roles distintos en el proceso de entrenamiento de modelos.

- `Parámetros`: son las variables internas del modelo que se ajustan durante el proceso de entrenamiento para que el modelo se adapte a los datos de entrenamiento y realice predicciones precisas. Por ejemplo, en un modelo de regresión lineal, los parámetros son los coeficientes y la intersección (Ver Figura 2.1). Los parámetros son características intrínsecas del modelo y son específicos del tipo de algoritmo utilizado. Se determinan automáticamente a partir de los datos de entrenamiento.

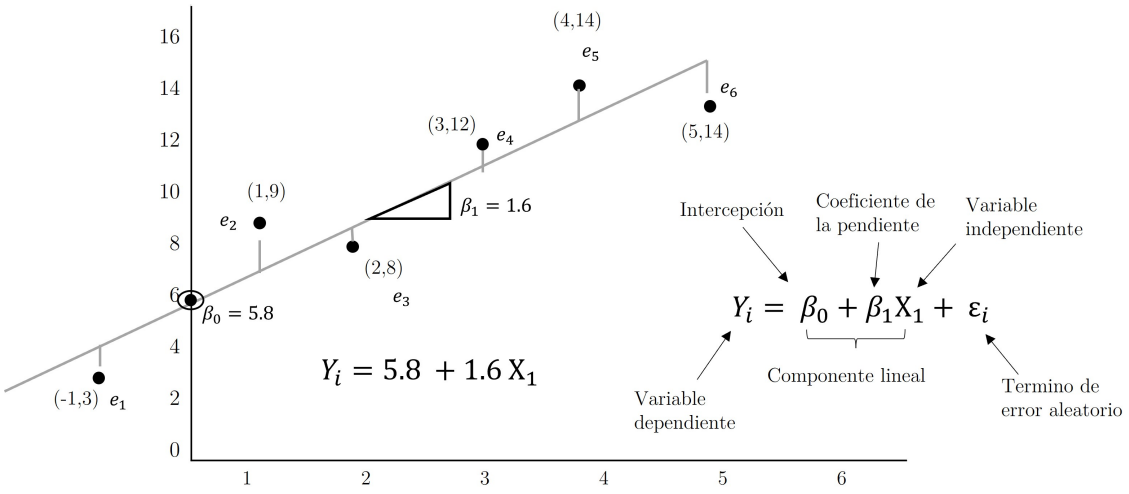


Figura 2.1: Parámetros en el modelo Regresión lineal.

- `Hiperparámetros`: son configuraciones ajustables del modelo que no se aprenden automáticamente a partir de los datos de entrenamiento. Deben establecerse antes de iniciar el proceso de entrenamiento y afectan cómo se entrena y se ajusta el modelo (ver Figura 2.2). Los hiperparámetros son configuraciones globales que controlan aspectos del proceso de entrenamiento, como la velocidad de aprendizaje, el número de épocas, la profundidad del árbol, la elección del *kernel* en máquinas de soporte vectorial, etc.

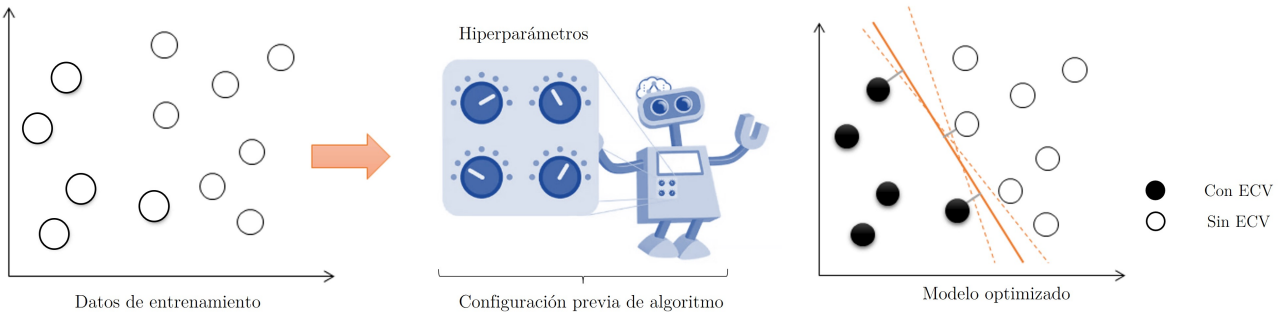


Figura 2.2: Etapa de la configuración de Hiperparámetros.

La relación que existe entre ambas variables radica en que los parámetros son características internas del modelo que se ajustan automáticamente, mientras que los hiperparámetros son configuraciones externas que afectan cómo se aprenden los parámetros.

2.1.2. Algoritmo Random Forest

El algoritmo Bosque Aleatorio, del inglés *Random Forest* (RF), es un enfoque de aprendizaje conjunto que utiliza un conjunto de árboles de decisión, cada uno de los cuales presenta pequeñas variaciones con respecto a los demás (Yang y Shami, 2020). El concepto básico de RF es que los árboles individuales proporcionan predicciones razonablemente precisas, pero con tendencia a sobreajustar ciertos puntos de datos. Al generar un gran número de árboles podemos reducir el sobreajuste promediando los resultados.

Sea X un subconjunto aleatorio de las observaciones y m un subconjunto aleatorio de las características, donde $m \leq p$. La clasificación de una nueva observación x se realiza votando los árboles individuales del bosque. La formulación matemática de un modelo RF puede escribirse como:

$$y = \text{Votación de la mayoría } \{T_1(x), T_2(x), \ldots, T_n(x)\} \tag{2.1}$$

$T_i(x)$ es la salida del árbol de decisión i para la observación x. La Figura 2.3 muestra la estructura general del algoritmo RF.

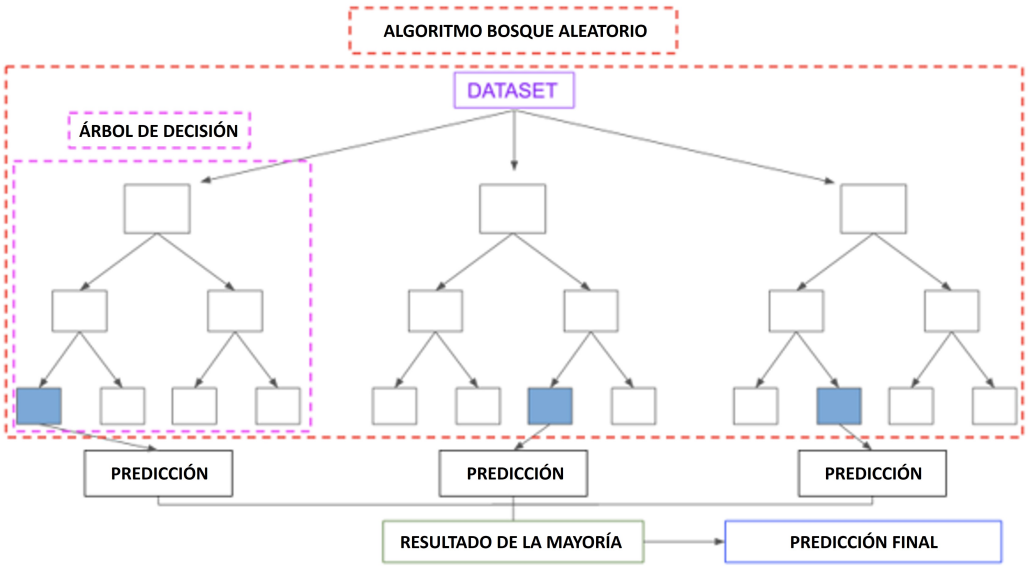


Figura 2.3: Estructura del algoritmo *Random Forest*.

El conjunto principal de hiperparámetros del algoritmo RF es el siguiente.

1. `n_estimators`: número de árboles de decisión considerados para entrenar el modelo.

2. `max_features`: número de características utilizadas para entrenar cada árbol.

3. `max_deph`: profundidad máxima de cada árbol de decisión.

4. `min_samples_leaf`: número mínimo de instancias necesarias en un nodo hoja.

5. `min_samples_split`: número mínimo de muestras necesarias para dividir un nodo interno en dos subnodos en cada árbol.

2.1.3. Algoritmo Support Vector Machine

El algoritmo Máquinas de Soporte Vectorial, del inglés *Support Vector Machine* (SVM), tiene como objetivo encontrar el hiperplano más favorable que distinga eficazmente las muestras pertenecientes a diferentes clases dentro de un espacio multidimensional. SVM se esfuerza por identificar el hiperplano que maximiza el margen entre estas clases, mejorando así su capacidad para manejar nuevos puntos de datos (Probst et al., 2018). Cuando se enfrenta a clases no linealmente separables, utiliza funciones de núcleo para transformar los datos en un espacio de mayor dimensión en el que se pueden distinguir. De este modo, el algoritmo puede clasificar nuevas instancias en función de su posición relativa al hiperplano y asignarlas con precisión a las clases predefinidas adecuadas.

Sea $\mathbf{x} \in X^p$ un vector de características y $z \in \{-1, 1\}$ la variable de respuesta binaria, el modelo SVM se define como (Yang y Shami, 2020):

$$g(x) = sng(\sum_{i=1}^{n} \lambda_i z_i \langle x_i^T x \rangle + a_0) \tag{2.2}$$

donde n es el número de observaciones en el conjunto de entrenamiento, λ_i son los coeficientes obtenidos por el modelo durante el entrenamiento, $\mathbf{x_i}$ son las características de la i-ésima observación, z_i es la variable de respuesta binaria para la i-ésima observación, $\langle \mathbf{x_i^T}, \mathbf{x}$ es el producto escalar entre los vectores de características $\mathbf{x_i}$ y $\mathbf{x}$, a_0 es el término de sesgo del modelo. La figura 2.4 muestra la estructura general del algoritmo SVM.

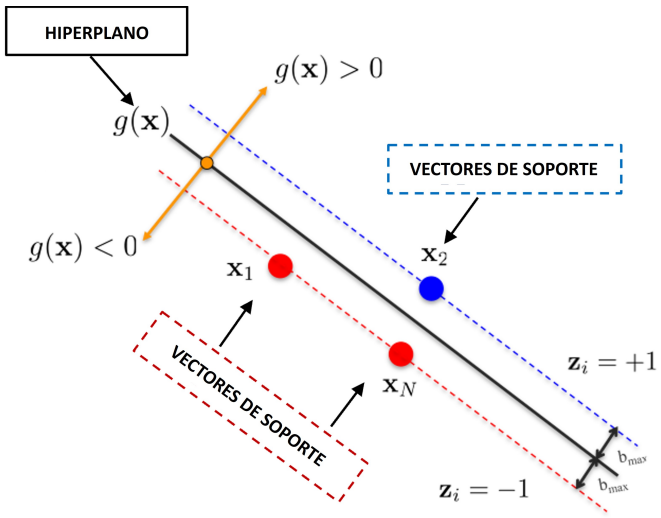


Figura 2.4: Estructura del algoritmo *Support Vector Machine*.

$g(x)$ utiliza una función *kernel* para medir la similitud entre dos puntos de datos x_i y x_j. Además, puede seleccionarse entre varios tipos de *kernels*. Las diferentes funciones de *kernel* se pueden denotar como sigue:

1. *Kernel* Lineal:

$$k(x, x^{'}) = x^T x^{'} \tag{2.3}$$

2. *Kernel* Polinomial:

$$k(x, x^{'}) = (\text{gamma} \quad x^T x^{'} + \text{coef0})^{\text{degree}} \tag{2.4}$$

3. *Kernel Radial Basis Function* (RBF):

$$k(x, x^{'}) = \exp(\text{- gamma} \quad ||x - x^{'}||^2) \tag{2.5}$$

4. *Kernel* Sigmoidal :

$$k(x, x^{'}) = \tanh(\text{gamma} \quad x^T x^{'} + \text{coef0}) \tag{2.6}$$

El conjunto de hiperparámetros más importantes es el siguiente.

1. `C`: El parámetro de regularización en `C` gestiona la ponderación entre maximizar el margen y minimizar el error de clasificación.

2. `kernel`: Determina el tipo de límite de decisión utilizado para separar las clases en el espacio de entrada.

3. `degree`: Determina la complejidad de la función de decisión (kernel) y, por tanto, la capacidad del modelo para ajustarse a los datos

4. `gamma`: Para el kernel RBF, controla la influencia de cada ejemplo de entrenamiento en el límite de decisión.

5. `coef0`: Para el kernel polinómico, determina el término independiente en la ecuación polinómica. Afecta a la no linealidad de la frontera de decisión.

2.1.4. Algoritmo XGBoost

El algoritmo *Extreme Gradient Boosting* (XGBoost) genera modelos secuenciales y utiliza la técnica de boosting para combinar árboles simples y menos precisos en árboles que mejoran la precisión de los casos predichos incorrectamente. El proceso de ajuste de cada árbol se realiza mediante el descenso de gradiente estocástico (Wu et al., 2019). El residuo se estima ajustando los datos a un árbol de decisión, y el segundo árbol se ajusta basándose en el residuo del paso anterior.

La función objetivo de XGBoost tiene un concepto de regularización que ayuda a seleccionar funciones predictivas y a controlar la complejidad del modelo. Combinando la función de pérdida y el término de regularización, se obtiene la función objetivo XGBoost. El poder predictivo del modelo se controla mediante la función de pérdida y la simplicidad del modelo se controla mediante el término de regularización. La función objetivo del XGBoost se muestra en la Ecuación 2.7 (Budholiya et al., 2022).

$$arg\ \min = \sum_{i=1}^{n} L(\hat{y}_i, y_i) + \sum_{i=1}^{k} R(f_i) \tag{2.7}$$

Donde L representa la función de pérdida que determina la compatibilidad del modelo n entrenar datos; etiqueta predicha se denota por y^i y y_i denota la etiqueta real. $R(f)$ se encarga de penalizar la complejidad de las funciones del árbol de entrenamiento. La representación del

algoritmo XGBoost se muestra en la Figura 2.5

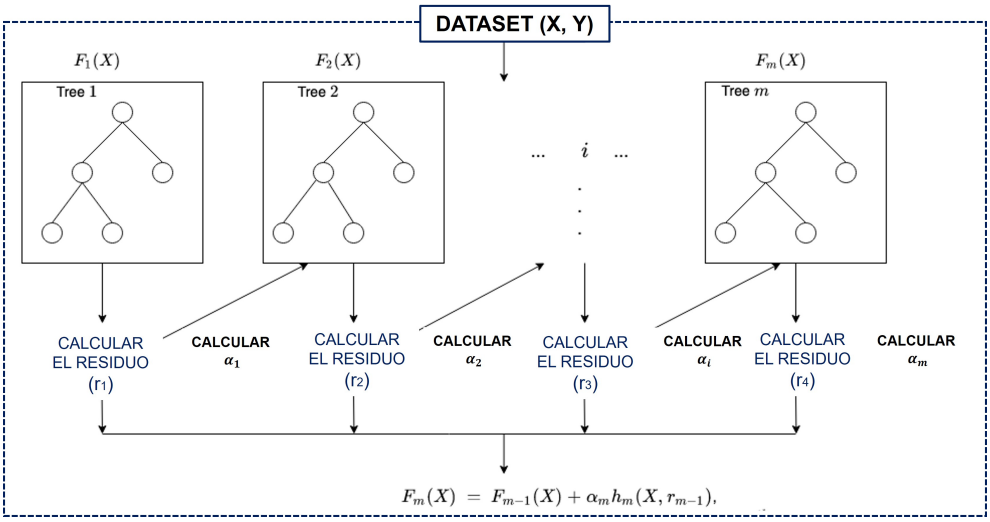


Figura 2.5: Estructura del algoritmo XGBoost.

Las variables (en la Figura 2.5) α_i y r_i son los parámetros de regularización y los residuos calculados con el árbol i^{th}, respectivamente, y h_i es una función entrenada para predecir los residuos r_i usando X para el árbol i^{th}. Para calcular α_i utilizamos los residuos r_i y calculamos lo siguiente:

$$arg\ \min_{\alpha} = \sum_{i=1}^{m} L(Y_i, F_{i-1}(X_i) + \alpha h_i(X_i, r_{i-1}) \tag{2.8}$$

donde $L(Y, F(X))$ es una función de costo diferenciable.

El conjunto de hiperparámetros del algoritmo XGBoost es el siguiente.

1. `n_estimators`: Número de árboles de decisión considerados para el entrenamiento del modelo.

2. `learning_rate`: Controla la velocidad a la que el modelo aprende de errores anteriores y ajusta los pesos de los árboles.

3. `gamma`: Controla la complejidad del modelo basado en árboles. Especifica la reducción mínima de pérdidas que debe alcanzarse para que un nodo se divida.

4. `subsample`: Determina la proporción de instancias a considerar para cada árbol. Permite muestrear aleatoriamente el conjunto de entrenamiento para cada árbol.

5. `max_depth`: Controla la profundidad máxima de cada árbol en el modelo.

6. `colsample_bytree`: Determina la proporción de características a considerar para cada árbol. Controla la aleatoriedad en la selección de características.

2.2. Métricas de evaluación en modelos de clasificación

Es importantes describir los cuatro tipos de valores (VP[1], VN[2], FN[3], FP[4]) que presentan los modelos de predicción de enfermedad cardiovascular antes de establecer las métricas que miden el desempeño de los modelos. Estos valores conforman la matriz de confusión que permite evaluar el desempeño de los modelos predictivos.

1. `Verdadero positivo`: se refiere a la situación en la que el valor real es positivo y el modelo también predice correctamente que es positivo..

2. `Verdadero negativo`: ocurre cuando el valor real es negativo y el modelo acierta al predecir que es negativo.

3. `Falso negativo`: indica que el valor real es positivo, pero el modelo erroneamente predice que es negativo.

4. `Falso positivo`: se presenta cuando el valor real es negativo, pero el modelo incorrectamente predice que es positivo.

Con los valores de predicción y algunas métricas podemos saber el rendimiento de nuestros modelos de forma cuantitativamente. Con la revisión de la literatura se identificaron las siguientes métricas para problemas de clasificación.

2.2.1 Exactitud

La métrica de exactitud, también conocida como *Accuracy*, se calcula como la proporción entre las predicciones correctas y el total de predicciones realizadas. Esta métrica proporciona una

[1]VP: Verdadero positivo
[2]VN: Verdadero negativo
[3]FN: Falso negativo
[4]FP: Falso positivo

evaluación global del rendimiento del modelo al medir la precisión general de sus predicciones.

$$\text{Accuracy} = \frac{VP + VN}{VP + FP + FN + VN} \tag{2.9}$$

2.2.2 Precisión

La métrica de precisión, también conocida como *Precision*, se define como la proporción entre las predicciones positivas verdaderas y el total de predicciones positivas realizadas por el modelo. Esta métrica evalúa la capacidad del modelo para identificar de manera correcta los casos positivos, destacando la proporción de instancias clasificadas como positivas que realmente lo son.

$$\text{Precision} = \frac{VP}{VP + FP} \tag{2.10}$$

2.2.3 Especificidad

La métrica especificidad , mejor conocida como *Specificity*, mide la proporción de verdaderos negativos identificados correctamente sobre el número total de negativos reales. Evalúa la capacidad del modelo para identificar correctamente los casos sin riesgo.

$$\text{Specificity} = \frac{VN}{VN + FP} \tag{2.11}$$

2.2.4 Sensibilidad

La métrica sensibilidad, mejor conocida como *Sensitivity/Recall*, representa la relación entre los casos positivos identificados correctamente y el número total de casos positivos reales. Mide la capacidad del modelo para encontrar todos los casos positivos.

$$\text{Recall} = \frac{VP}{VP + FN} \tag{2.12}$$

2.2.5 Puntaje F1

La métrica puntaje F1, mejor conocida como *F1-Score*, combina la métrica *Precisión* y *Sensibilidad* en una única métrica que representa el equilibrio entre ambas. Resulta útil cuando se desea tener en cuenta tanto la precisión como la sensibilidad en la evaluación de modelos.

$$\text{F-Score} = \frac{2 \times Precision \times Sensibilidad}{Precision + Sensibilidad} \tag{2.13}$$

2.3. Configuración de hiperparámetros como problema de optimización

2.3.1. Optimización Matemática

Un problema de optimización implica la búsqueda de valores para ciertas variables, llamadas variables de decisión. Estos valores deben cumplir con un conjunto de restricciones representados por ecuaciones y/o inecuaciones algebraicas, conocidas como restricciones, que limitan las opciones para las variables de decisión. El objetivo es encontrar los valores de las variables de decisión que optimizan una función objetivo. Esta función se utiliza para medir el rendimiento del sistema en estudio, buscando alcanzar el mejor valor posible. En el trabajo de Yang y Shami, 2020 se plantea que los problemas de optimización pueden clasificarse como problemas con o sin restricciones. La mayoría de los problemas de optimización de la vida real contienen un conjunto de restricciones. La variable de decisión x para problemas de optimización con restricciones debe estar sujeta a ciertas restricciones que pueden ser igualdades o desigualdades matemáticas.

Espinosa-Paredes y Rodríguez, 2016 define un problema de optimización como un problema de decisión. En este contexto, se busca determinar valores para un conjunto de variables, las cuales están relacionadas mediante expresiones matemáticas. Estos valores deben minimizar o maximizar una función objetivo diseñada para cuantificar el rendimiento y evaluar la calidad de la decisión. Por lo general, se consideran restricciones que limitan las opciones para las variables, y el objetivo es encontrar la configuración óptima que cumpla con dichas restricciones.

En resumen, un problema de optimización puede expresarse como:

$$\text{mín} \,/\, \text{máx}_x f\,(x) \tag{2.14}$$

con respecto a:

$$g_i\,(x) \leq 0, i = 1, 2, \ldots, m.$$

$$h_i\,(x) = 0, j = 1, 2, \ldots, p.$$

$$x \ \in X$$

Los componentes de un problema de optimización son (Bradley et al., 1977):

- `Variable de decisión`: estas variables describen los posibles diseños candidatos y las condiciones de funcionamiento del sistema en cuestión. Se eligen como variables independientes aquellas que ejercen un impacto significativo sobre la función objetivo.
- `Función objetivo`: esta función desempeña un papel fundamental en la toma de decisiones al determinar los valores óptimos de las variables de decisión que resuelven el problema de optimización. La función objetivo permite identificar los valores más apropiados para las variables de decisión.
- `Restricciones`: son ecuaciones o inecuaciones que definen las relaciones entre las variables de decisión. Estas relaciones surgen debido a diversas razones, como limitaciones en el sistema, leyes naturales o restricciones tecnológicas.

El objetivo de las restricciones es limitar los posibles valores de la solución óptima a ciertas áreas del espacio de búsqueda, denominadas región factible. Así, la región factible D de x puede ser representada como:

$$R_f = \{x \in X \mid g_i(x) \ \leq 0, \ h_i\,(x) = 0\} \tag{2.15}$$

Es el conjunto de todos los puntos posibles de un problema de optimización que satisfacen las restricciones del problema (Espinosa-Paredes y Rodríguez, 2016). En los problemas de

optimización hay que distinguir dos tipos de soluciones; es común describirlos en términos de optimización local y optimización global. Un óptimo global es una solución para el problema global de optimización. El valor objetivo es tan bueno como cualquier otro punto de R_f. Un óptimo local, en cambio, es óptimo sólo con respecto a las soluciones factibles cercanas a ese punto de R_f (Bradley et al., 1977).

2.3.2. Formalización del proceso de optimización de hiperparámetros

Sea y una variable objetivo, un vector de características X, y una distribución desconocida P sobre (X, y), de la que se ha muestreado un conjunto de datos T de n observaciones. Un algoritmo M aprende la relación funcional entre X e y produciendo un modelo predictivo $f(X, \theta)$, controlado por la configuración k-dimensional de hiperparámetros $\theta = (\theta_1, \ldots, \theta_k)$ del espacio de búsqueda de hiperparámetros $\Theta = \Theta_1 \times \ldots \times \Theta_k$. Para medir el rendimiento de la predicción entre la etiqueta verdadera y y su predicción $f(X, \theta)$, definimos una función de costo $L(y, f(X, \theta))$.

El objetivo es estimar el riesgo esperado de M, con respecto a θ en los nuevos datos, también muestreados de $P : R(\theta) = E(L(y, f(X, \theta))|P)$. Esta asignación codifica, dada una distribución de datos específica, un algoritmo de aprendizaje y una medida de rendimiento específica, la calidad numérica para cualquier configuración de hiperparámetros θ.

Dadas m distribuciones de datos diferentes $P_1, \ldots P_m$, se obtienen m hiperparámetros mapeados (Probst et al., 2018):

$$R_j(\theta) := E(L(y, f(X, \theta))), \quad j = 1, \ldots, m \tag{2.16}$$

El proceso de optimización de hiperparámetros consiste en encontrar la configuración de hiperparámetros que mejore el proceso de aprendizaje de los modelos, aplicando enfoques de optimización. Cada enfoque trata la configuración de hiperparámetros como un problema de optimización, donde la función objetivo es minimizar los errores de predicción de los modelos, tratando los hiperparámetros como variables de decisión. Así, la mejor configuración de hiperparámetros para el conjunto de datos j se define como (Pannakkong et al., 2022):

$$\theta^* := \arg\min_{\theta \in \Theta} R^j(\theta) \tag{2.17}$$

La función objetivo $f(x)$ determina el riesgo del modelo de aprendizaje automático cuando es entrenado con un conjunto de datos j. Esta función compara las predicciones del modelo con la clase real para determinar la tasa de error del algoritmo. Al tratarse de una formulación matemática, la función de costo expresa numéricamente el nivel de error de predicción.

2.3.3. Componentes y fases del proceso de optimización

Yang y Shami, 2020 establecen que un proceso de optimización de hiperparámetros debe tener los siguientes cuatro componentes:

1. Un estimador que puede ser un regresor o clasificador: en nuestro caso seleccionamos los clasificadores RF, SVM y XGBoost.

2. Espacio de búsqueda para el conjunto de hiperparámetros: por cada algoritmo de aprendizaje automático estamos proponiendo el rango de valores para cada hiperparámetros que se optimizará.

3. Enfoque de optimización para determinar las combinaciones. En nuestro caso aplicaremos: *Grid Search, Random Search, Bayesian Optimization, Particle Swarm Optimization y Genetic Algorithm.*

4. Una métrica que evalúe el desempeño de los modelos, con ella se hará una comparación entre las distintas combinaciones que se han evaluado. Se seleccionó la métrica ROC_AUC.

2.4. Enfoques de optimización de hiperparámetros

En las últimas décadas, se han desarrollado varios enfoques optimización de hiperparámetros para facilitar y automatizar la búsqueda de combinaciones óptimas para problemas de configuración de hiperparámetros. Sin embargo, los métodos de optimización no son tan utilizados en la práctica como deberían debido a los siguientes factores (Bischl et al., 2023).

- Los usuarios potenciales pueden tener dificultades para comprender los métodos de optimización de hiperparámetros, percibiéndolos como cajas negras demasiado complejas.
- Falta de confianza en la superioridad de los métodos de optimización de hiperparámetros frente a un enfoque manual o basado en experiencia, lo que genera dudas sobre su rendimiento esperado.
- Falta de conocimientos sobre cómo seleccionar y configurar adecuadamente los métodos de optimización en función del problema planteado.
- Incertidumbre sobre la definición correcta del espacio de búsqueda de hiperparámetros para el método de optimización

A continuación, se describen los enfoques de configuración de hiperparámetros empleados en esta investigación.

2.4.1 Configuración predeterminada o manual

Cuando se construye un modelo de aprendizaje automático, es común que los hiperparámetros se utilicen en sus valores predeterminados, que son generalmente seleccionados por los desarrolladores de las bibliotecas de aprendizaje automático. Sin embargo, estos valores predeterminados pueden no ser los más adecuados para un conjunto de datos y un problema específico (Yang y Shami, 2020).

Por otro lado, en el enfoque manual, un experto en el dominio o un científico de datos ajusta los hiperparámetros del modelo de manera iterativa y basada en su conocimiento y experiencia. Sin embargo, este método tiene los siguientes desafíos y limitaciones (Probst et al., 2018):

- `Subjetividad`: la selección de hiperparámetros depende en gran medida de la intuición y experiencia del experto. Esto puede llevar a decisiones sesgadas o subóptimas.
- `Tiempo y Recursos`: el ajuste manual puede ser un proceso lento y costoso en términos de recursos humanos. Probar múltiples combinaciones de hiperparámetros puede requerir un esfuerzo considerable.

- `Exploración Limitada`: debido a restricciones de tiempo, los expertos pueden no explorar exhaustivamente todas las combinaciones posibles de hiperparámetros, lo que podría resultar en la omisión de configuraciones prometedoras.

- `Solución local`: a pesar del esfuerzo invertido, el enfoque manual no garantiza encontrar la configuración de hiperparámetros óptima para un conjunto de datos y problema específicos.

2.4.2 Grid Search

La búsqueda en rejilla, del inglés *Grid Search* (GS) se utiliza ampliamente en el aprendizaje automático para descubrir la combinación más eficaz de los hiperparámetros de un modelo. Este método consiste en especificar un conjunto de valores posibles para cada hiperparámetro y, a continuación, realizar una búsqueda exhaustiva en una rejilla multidimensional para evaluar el rendimiento del modelo para cada combinación de hiperparámetros (Andonie, 2019).

Es fácil ejecutar GS en paralelo porque cada ensayo se ejecuta individualmente y el resultado es independientes de los de otros ensayos. Sin embargo, GS sufre la maldición de la dimensionalidad porque el consumo de recursos informáticos aumenta exponencialmente cuando hay hiperparámetros. Por otro lado, un rango de muestreo limitado es aceptable para GS porque no es deseable que haya demasiadas configuraciones (Prabu et al., 2022). Una gran desventaja es que no detecta el óptimo global de los parámetros continuos, ya que requiere un conjunto predefinido y finito de valores de hiperparámetros.

En la Figura 2.6 se muestra forma generalizada el mecanismo de búsqueda de GS.

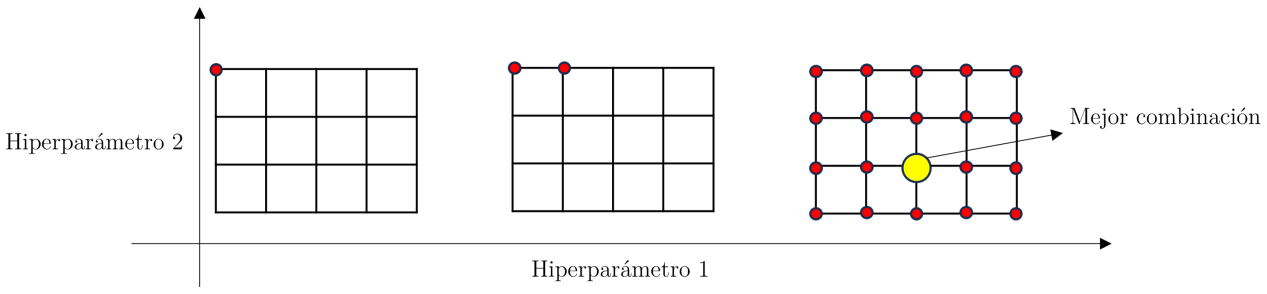


Figura 2.6: Enfoque búsqueda en rejilla.

El procedimiento que sigue de GS se describe en el Algoritmo 1.

Algoritmo 1 Búsqueda en Rejilla

Data:
Espacio de búsqueda de hiperparámetros Θ
Modelo de aprendizaje automático M
Conjunto de datos de entrenamiento D_{train}
Result: Mejor configuración de hiperparámetro θ^*
$\theta^* \leftarrow \emptyset$
$best\_performance \leftarrow -\infty$
for θ *in* Θ **do**
  $M_\theta \leftarrow$ Inicializar el modelo M con los hiperparámetrosθ
  Entrenar M_θ con D_{train}
  $performance \leftarrow$ Evaluar el desempeño de M_θ con el conjunto de prueba
  if $performance > best\_performance$ **then**
    $best\_performance \leftarrow performance$
    $\theta^* \leftarrow \theta$
  end
end

2.4.3 Random Search

La búsqueda aleatoria, del inglés, *Random Search* (RS) surgió como solución para superar ciertas limitaciones del enfoque GS. Aunque comparte similitudes con GS, este enfoque en lugar de probar todos los valores del espacio de búsqueda selecciona aleatoriamente un número predefinido de muestras dentro de los límites superior e inferior como valores candidatos para los hiperparámetros. Estos candidatos se entrenan hasta que se agota el presupuesto definido. La idea de RS es que, si el espacio de configuración es lo suficientemente grande, es posible encontrar óptimos globales, o al menos aproximaciones a ellos (Bergstra y Bengio, 2012). El procedimiento que sigue RS se describe en el Algoritmo 2 y su mecanismo en la Figura 2.7.

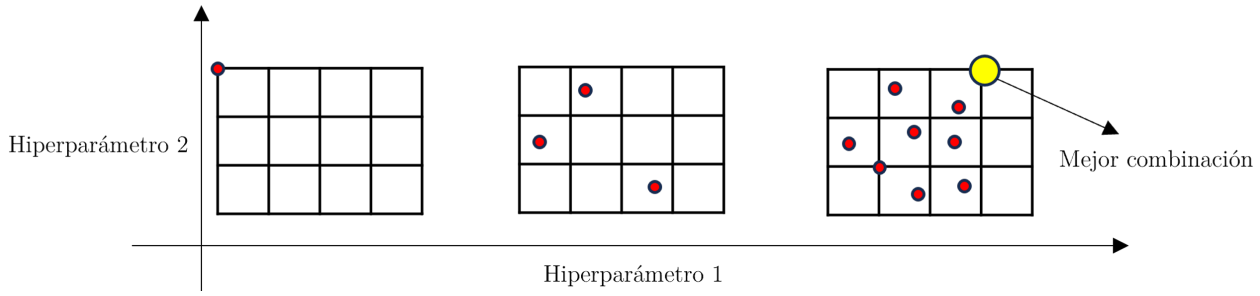


Figura 2.7: Enfoque búsqueda aleatoria.

Algoritmo 2 Bayesian Optimization

Data:
Espacio de búsqueda de hiperparámetros Θ
Modelo de aprendizaje automático M
Conjunto de datos de entrenamiento D_{train}
Número de iteraciones n
Result: Mejor configuración de hiperparámetro θ^*
$\theta^* \leftarrow \emptyset$
$best\_performance \leftarrow -\infty$
for $i \leftarrow 1$ *to* n **do**
  $\theta \leftarrow$ Muestreo aleatoreo de Θ
  $M_\theta \leftarrow$ Inicializar el modelo M con los hiperparámetrosθ
  Train M_θ with D_{train}
  $performance \leftarrow$ Evaluar el desempeño de M_θ con el conjunto de prueba
  if $performance > best\_performance$ **then**
    $best\_performance \leftarrow performance$
    $\theta^* \leftarrow \theta$
  end
end

2.4.4 Bayesian Optimization

Plantear un problema de optimización de hiperparámetros en modelos de aprendizaje automático puede caracterizarse como una función de caja negra computacionalmente compleja, ya que requiere una inversión de tiempo significativa y carece de una formulación matemática transparente que establezca una correlación directa entre los hiperparámetros y la precisión del modelo. Además, la evaluación de la eficacia de un conjunto de hiperparámetros requiere la repetición del entrenamiento del modelo y su posterior evaluación en un conjunto de validación,

lo que supone un esfuerzo no trivial (Luong et al., 2019).

La optimización bayesiana, del inglés *Bayesian Optimization* (BO) es un enfoque para identificar globalmente soluciones óptimas dentro de complicadas funciones de caja negra que no son convexas por naturaleza, requieren una evaluación intensiva en recursos y carecen de un marco analíticamente solucionable para calcular derivadas (Wu et al., 2019). Mediante el uso de datos observados para anticipar el siguiente punto de evaluación, BO demuestra la capacidad de identificar la configuración óptima de hiperparámetros en un número modesto de pasos iterativos (Hazan et al., 2017).

El objetivo general de BO es encontrar la configuración óptima x^* que maximiza el valor de la función $f(x)$ utilizando una función objetivo desconocida f que recibe un valor de entrada x, donde la función objetivo real es desconocida. Sin embargo, necesitamos dos cosas para examinar los valores de la función secuencialmente para los candidatos de valor de entrada y encontrar la configuración óptima que maximiza $f(x)$ (Kim y Chung, 2019):

1. El primero es un modelo sustituto que realiza una estimación probabilística de la naturaleza de la función objetivo desconocida basada en los valores de entrada y los valores de función explorados hasta el momento.

2. El segundo consiste en una función de adquisición que deriva el valor óptimo de entrada x^* basado en los resultados de la estimación probabilística hasta el momento.

En este trabajo utilizaremos un Proceso Gaussiano (GP) como modelo sustituto, pero existen otras alternativas como Random Forest y Tree-structured Parzen Estimators (TPE) (Bergstra y Bengio, 2012). Los GPs proporcionan modelos para distribuciones gaussianas, así como para otras variables aleatorias de uso común en estadística. El modelo correspondiente se muestra en la siguiente ecuación:

$$f(x) \sim gp(m(x),\ k(x, x^{'})) \tag{2.18}$$

En el contexto del enfoque BO, la función f se modela como una GP con una función media m y una función de covarianza k. Cuando se aplica BO, es común simplificar m a cero, y la elección utilizada con frecuencia para k es el núcleo exponencial cuadrado, como se menciona en (Luong et al., 2019).

$$k(x, x^{'}) = \sigma^2 exp(-\frac{1}{2l^2}||x - x^{'}||^2) \tag{2.19}$$

donde σ^2 es un parámetro que dicta la incertidumbre en $f(x)$ y l es un parámetro de escala de longitud que controla la rapidez con que una función puede cambiar.

Sea $D = \{(x_i, y_i)\}$ las observaciones que contienen N entradas x_i y sus correspondientes valores de función $y_i = f(x_i) + \epsilon_i$ y $\epsilon_i \sim N(0, \sigma_\epsilon^2)$. Ajustando los datos observados a la GP, obtenemos la distribución predictiva de $f(x)$ en cualquier punto x del espacio de búsqueda. La distribución predictiva es también una un GP caracterizado por la media y la varianza como sigue (Luong et al., 2019):

$$m(x) = k^T(K + \sigma_\epsilon^2 I)^{-1} y \tag{2.20}$$

$$\sigma^2(x) = k(x, x^{'}) - k^T(K + \sigma_\epsilon^2 I)^{-1} k \tag{2.21}$$

donde $y = (y_1, \ldots, y_N)$ es un vector de los valores de la función que tenemos hasta ahora, $k(x, x)$ es la covarianza en el punto x, $k = [k(x_i, x)] \forall x_i \in D$ es la covarianza entre el nuevo punto x y todos los demás puntos observados $x_i, K = [k(x_i, x_j) \forall x_i, x_j \in D]$ es la matriz de covarianza, I es una matriz de identidad con la misma dimensión que K, y $\sigma_\in^2$ es el ruido de medición.

La media y la varianza posteriores, calculadas mediante las ecuaciones 2.20 y 2.21 respectivamente, se emplean para formular una función de adquisición denominada $\alpha(x)$. En los trabajos de Brochu et al., 2010; Mockus et al., 1978 se analizan distintas variaciones de las funciones de adquisición.

Los pasos se describen al enfoque BO se presentan en el Algoritmo 3.

Algoritmo 3 Optimización Bayesiana

Data:
Proceso gaussiano a priori sobre f.
Observar f en n_0 puntos de acuerdo con un diseño experimental inicial de llenado de espacio.
Inicializar $n = n_0$.
while $n \leq N$ **do**
  Actualizar la distribución de probabilidad posterior sobre f utilizando todos los datos disponibles.
  Sea x_n un maximizador de la función de adquisición sobre x, donde la función de adquisición se calcula utilizando la distribución posterior actual.
  Obsérvese que $y_n = f(x_n)$.
  Incrementar n
end
Result: ya sea el punto evaluado con la mayor $f(x)$, o el punto con la mayor media posterior.

2.4.5 Particle Swarm Optimization

El algoritmo Optimización por Enjambre de Partículas, del inglés *Particle Swarm Optimization* (PSO) fue propuesto por James Kennedy y Russell Eberhart en 1995 (Kennedy y Eberhart, 1995). PSO se utiliza para encontrar el máximo o mínimo de una función definida en un espacio vectorial multidimensional. Consiste en colocar cierto número de partículas (también llamado población) en el espacio de búsqueda de la función a optimizar.

El enjambre opera mediante los desplazamientos de las partículas, las cuales se sitúan en un espacio de búsqueda asociado a un problema o función específica. Cada partícula, en su posición actual, evalúa la función objetivo correspondiente. La posición de una partícula representa una solución concreta, mientras que la velocidad rige la dirección y la magnitud de su movimiento dentro del espacio de soluciones. Mediante actualizaciones iterativas, las partículas exploran y explotan en colaboración el espacio de búsqueda, garantizando que el algoritmo converja a soluciones favorables para el problema de optimización planteado (Yang y Shami, 2020).

Según la investigación de Morales López y Labrín, 2010, el movimiento de las partículas en su espacio de búsqueda se determina mediante la combinación de varios aspectos de su historial actual y las mejores ubicaciones de uno o más miembros del enjambre, todo ello sujeto a ciertas perturbaciones aleatorias. En otras palabras, una partícula ajusta su velocidad con el objetivo de encontrar una solución óptima (posición) al incorporar su propia experiencia (memoria, es decir, teniendo en cuenta sus mejores posiciones en iteraciones anteriores) y la experiencia de

las partículas vecinas (es decir, las mejores soluciones encontradas en la población).

Cada partícula en el enjambre se compone de dos vectores de largo D, lo cual hace referencia a la dimensión del espacio de búsqueda.

1. `Vector de posición`: representa la solución propuesta por la partícula.

$$\vec{x^t} = [x_1, x_2, \ldots, x_D]^T \tag{2.22}$$

2. `Vector de velocidad`: representa la velocidad con la que la partícula se mueve en el espacio de búsqueda.

$$\vec{V^t} = [v_1, v_2, \ldots, v_D]^T \tag{2.23}$$

Las ecuaciones utilizadas para actualizar la posición y la velocidad de una partícula i en el espacio de búsqueda en la iteración t se dan en las ecuaciones 2.24 y 2.25, respectivamente (Kennedy y Eberhart, 1995).

$$X_i^{(t+1)} = X_i^t + V_i(t+1) \tag{2.24}$$

$$V_i^{(t+1)} = V_i^t w + c_1 r_1 (y_i - X_i^t) + c_2 r_2 (\hat{y} - X_i^t) \tag{2.25}$$

Donde V_{id}^t es la velocidad de la partícula i. V_{id}^{t+1} es la velocidad actualizada de la partícula i en la iteración $t+1$; w es una constante de amplitud de inercia, reflejando la importancia de la velocidad previa; X_i^t representa la posición de la partícula i. En cada iteración del algoritmo, la posición actual X_i^t se evalúa como una solución al problema. Si la posición es mejor que cualquier otra hasta ese momento, se almacena en el vector y_i. X_i^{t+1} es la posición actualizada de la partícula i; y_i es la mejor solución local (*pbest*) que representa la mejor experiencia personal de la partícula. Esta variable es utilizada como comparación durante el proceso de búsqueda; $\hat{y}$ es la mejor solución global (*gbest*) y representa la mejor solución encontrada por la población de partículas; c_1 y c_2 son las constantes de aceleración que influyen en la velocidad de las partículas hacia la mejor posición.; r_1 y r_2 son números aleatorios en el rango $[0, 1]$, imitando el comportamiento impredecible de un enjambre y aportando un componente estocástico al

algoritmo.

Es importante mencionar que en la versión canónica (Kennedy y Eberhart, 1995) se limitaban las velocidades dentro del rango $[-V_{max}, +V_{max}]$, con la finalidad de que las partículas no tuvieran movimientos muy bruscos y evitar que pudiesen salir del espacio de búsqueda (Morales López y Labrín, 2010).

Para calcular la aptitud de una partícula en una tarea de minimización, se emplea una función objetivo f, considerando que el enjambre está compuesto por n partículas. La actualización de los mejores valores global y personal en la iteración t se realiza mediante las ecuaciones 2.26 y 2.27, respectivamente.

$$\text{gbest} = \arg\min_{i=1}^{N} \{f(\text{pbest}_i)\} \tag{2.26}$$

$$\text{pbest}_i(t+1) = \begin{cases} \text{pbest}_i(t) & : f(\text{pbest}_i(t)) \leq f(p(t+1)) \\ X_i^{(t+1)} & : \text{En caso contrario} \end{cases} \tag{2.27}$$

Donde *gbest* es la mejor posición global encontrada por todas, N es el número total de partículas del enjambre, $pbest_i$ es la mejor posición individual de la partícula i, $f(x)$ es la función objetivo que se optimiza y $\arg\min_{i=1}^{N}$ representa el índice de la partícula cuya mejor posición $pbest_i$ arroja el valor mínimo de la función objetivo. El procedimiento PSO se describe en el Algoritmo 4.

Algoritmo 4 Optimización por Enjambre de Partículas

Data:
Función de aptitud: f
Factor cognitivo: C_{cog}
Factor social: C_{soc}
Result: gbest
Inicialización:
for *cada particula i en el enjambre* **do**
  Inicializar la posición: $\mathbf{X}_i \leftarrow$ posición aleatoria en el espacio de búsqueda;
  Inicializar la velocidad: $\mathbf{V}_i \leftarrow$ velocidad aleatoria en el espacio de búsqueda;
  Evaluar la aptitud de la partícula: $f(\mathbf{X}_i) \leftarrow$ valor de la función de aptitud de $\mathbf{X}_i$;
  Establece tu mejor posición personal: $\mathbf{pbest}_i \leftarrow \mathbf{X}_i$
  if $f(\mathbf{X}_i) < f(\mathbf{gbest})$ **then**
    Actualizar la mejor posición global: $\mathbf{gbest} \leftarrow \mathbf{X}_i$
  end
end
Actualizar posiciones y velocidades:
while *no se cumple el criterio de rescisión* **do**
  for *cada particula i en el enjambre* **do**
    Actualiza la velocidad de las partículas: $\mathbf{V}_i \leftarrow \mathbf{V}_i\omega + c_{\text{cog}} r_1(\mathbf{pbest}_i - \mathbf{X}_i) + c_{\text{soc}} r_2(\mathbf{gbest} - \mathbf{X}_i)$
    Velocidad límite: $\mathbf{V}_i \leftarrow \text{clip}(\mathbf{V}_i, v_{\min}, v_{\max})$
    Actualiza la posición de la partícula: $\mathbf{X}_i \leftarrow \mathbf{X}_i + \mathbf{V}_i$
    Evaluar la aptitud de la nueva posición: $f(\mathbf{X}_i) \leftarrow$ valor de la función de aptitud de $\mathbf{X}_i$
    if $f(\mathbf{X}_i) < f(\mathbf{pbest}_i)$ **then**
      Actualizar la mejor posición personal: $\mathbf{pbest}_i \leftarrow \mathbf{X}_i$
      if $f(\mathbf{X}_i) < f(\mathbf{gbest})$ **then**
        Actualizar la mejor posición global: $\mathbf{gbest} \leftarrow \mathbf{X}_i$
      end
    end
  end
end

2.4.6 Genetic Algorithm

El Algoritmo Genético, del inglés *Genetic Algorithm* (GA) es una técnica de búsqueda estocástica basada en el mecanismo de la selección natural. La idea de GA, como algoritmo inspirado en la teoría de la evolución, fue presentada por John Holland en la década de 1960, y luego desarrollada por David E. Goldberg en la década de 1980 (Jaramillo et al., 2002). Dado un problema específico a resolver, la entrada del AG es un conjunto de soluciones potenciales a ese problema, codificadas de alguna manera, y una métrica llamada función de aptitud, o

fitness, que permite evaluar cuantitativamente a cada solución candidata (Nayyar et al., 2018).

A diferencia de las técnicas de búsqueda convencionales (GS o RS), el enfoque AG parte de un conjunto inicial de soluciones, denominado población. Cada miembro de la población se denomina cromosoma, que en nuestro contexto particular representa una solución potencial al problema dado, es decir, un modelo que contiene una configuración de hiperparámetros. Un cromosoma consiste en una secuencia de símbolos, típicamente pero no exclusivamente representados como bits binarios. Estos cromosomas pasan por iteraciones sucesivas, llamadas generaciones. Durante cada generación, los cromosomas se evalúan utilizando ciertas métricas de aptitud (Jaramillo et al., 2002). Para formar la siguiente generación, se crean nuevos cromosomas, llamados descendientes, mediante (Bischl et al., 2023):

- dos cromosomas de la generación actual mediante una operación de cruce, y
- Introducir cambios en un cromosoma utilizando una operación de mutación.

En la Figura 2.8 se ilustra la terminología que se utiliza para describir los elementos fundamentales que componen el proceso evolutivo del algoritmo genético de describe a continuación.

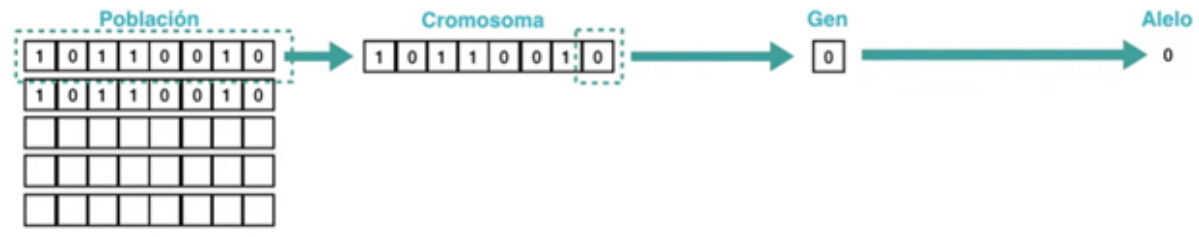


Figura 2.8: Elementos fundamentales que componen al enfoque GA.

- `Población`: es un conjunto de individuos o soluciones candidatas que se utilizan para encontrar una solución óptima o cercana a ella en un problema dado.
- `Cromosoma`: es una representación de una solución candidata en el espacio de búsqueda. Similar a cómo los cromosomas contienen genes en la genética biológica, los cromosomas en algoritmos genéticos contienen genes que representan características o variables del problema en cuestión.

- `Gen`: es una parte indivisible de un cromosoma que representa una característica o variable específica de una solución candidata. Cada gen dentro del cromosoma codifica un valor o una configuración para esa característica particular. Durante el proceso de reproducción en algoritmos genéticos, los genes se combinan mediante operadores de cruzamiento (*crossover*).

- `Alelo`: se refiere a las diferentes variantes o opciones que un gen puede tener. Cada alelo representa una posible configuración o valor que puede tomar el gen en cuestión durante el proceso de reproducción.

La siguiente generación se deriva de esta población intermedia: i) seleccionando determinados progenitores y descendientes en función de sus valores de aptitud, y ii) excluyendo a otros para mantener un tamaño de población constante. Los cromosomas con valores de aptitud más altos tienen más probabilidades de ser seleccionados (ver Figura 2.9). Después de varias generaciones, se espera que la solución óptima o subóptima converja, representando el mejor resultado posible para el problema dado (Jaramillo et al., 2002).

GA es fácil de implementar y no necesita buenas inicializaciones, ya que sus operaciones de selección, cruce y mutación reducen la posibilidad de no alcanzar el óptimo global. Sin embargo, la principal limitación de GA es que el propio algoritmo introduce parámetros adicionales que deben configurarse, incluidos el tipo de función de aptitud, el tamaño de la población, la tasa de cruce y la tasa de mutación (Yang y Shami, 2020).

El pseudocódigo del algoritmo genético clásico se muestra en el Algoritmo 5.

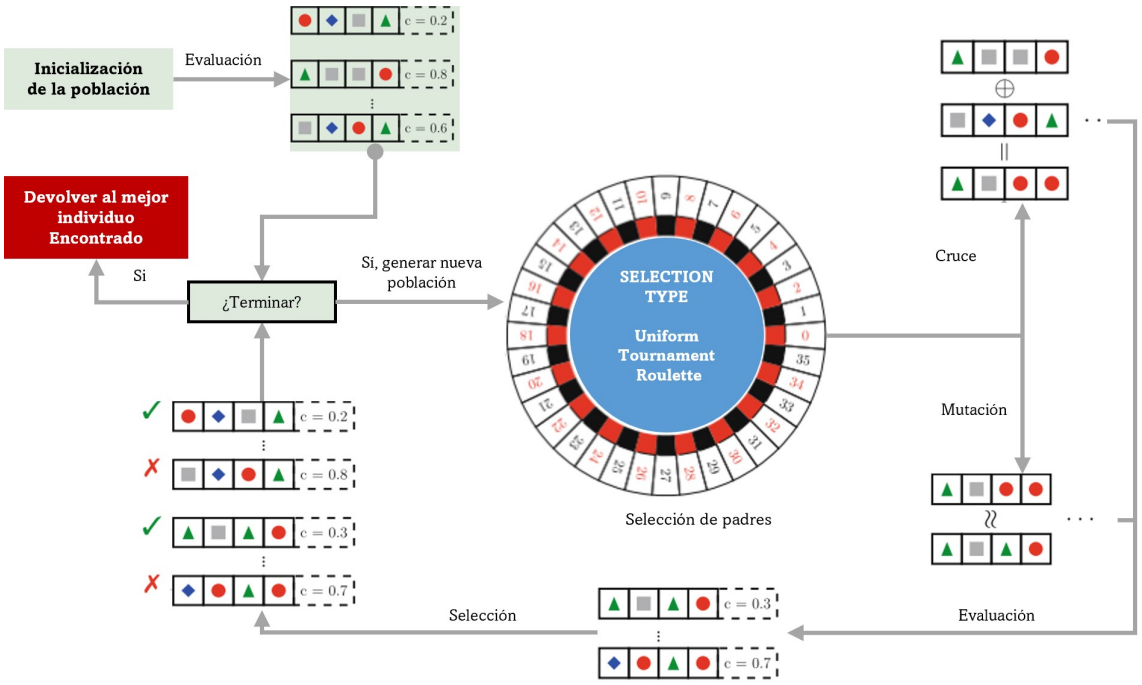


Figura 2.9: Esquema del algoritmo genético como problema discreto de cuatro dimensiones (figura inspirada en Bischl et al., 2023.)

Algoritmo 5 Enfoque Genético

Data:
Población de individuos
Inicializar la población P con individuos aleatorios
Establecer g = 1
Crear una población inicial
while $g \leq N\_generations$ **do**
  for *cada individuo i de la población P* **do**
    Calcular la aptitud: $f_i = \text{Fitness}(i)$
  end
  Seleccionar a los padres para la reproducción mediante un mecanismo de selección
  for *cada pareja de padres* (p_1, p_2) **do**
    Realizar cruces y mutaciones para crear descendencia
    Calcula la aptitud de la descendencia: $f_\beta = \text{Fitness}(\beta)$
  end
  Seleccionar individuos para la siguiente generación utilizando la estrategia de reemplazo
  Incrementar g
end
Return: Mejor solución (población)

2.5. Enfermedades cardiovasculares

Las enfermedad cardiovasculares son condiciones que afectan la estructura y el funcionamiento del corazón. En términos generales, los factores de riesgo más importantes son: consumo de tabaco, consumo nocivo de alcohol, colesterol de la sangre, diabetes, cifras elevadas de presión arterial, obesidad, falta de actividad física, antecedentes familiares con esta enfermedad y el estrés (Texas-Heart-Institute, 2022). A nivel mundial, son la principal causa de muerte debido a diferentes factores; la inactividad física, malos hábitos de alimentación, alto nivel de colesterol, hipertensión y estrés son las principales (Jain y Singh, 2018). Cada año se pierden 17.9 millones de vidas debido a estas enfermedades. Además, se estima que en 2030 cerca de 23.6 millones de personas tendrán algún padecimiento (Purushottam et al., 2016).

A continuación, se describen los tipos de enfermedades cardiovasculares más comunes.

1. `Arritmia`: según la British-Heart-Foundation, 2017 un ritmo cardíaco anormal se denomina arritmia. El corazón tiene un sistema de conducción eléctrica que bombea sangre por todo el cuerpo. Las arritmias están causadas por una anomalía en ese sistema de conducción eléctrica y pueden hacer que el corazón lata demasiado lento, demasiado rápido o de forma irregular.

2. `Insuficiencia cardíaca`: es una afección en la cual el corazón ya no puede bombear sangre rica en oxígeno al resto del cuerpo de forma eficiente. Esto provoca que se presenten síntomas en todo el cuerpo (MedlinePlus, 2022b). La insuficiencia cardíaca por lo regular es una afección prolongada (crónica), pero se puede presentar repentinamente.

3. `Hipertensión`: es la fuerza que ejerce la sangre contra las paredes de las arterias. Se considera que la persona presenta hipertensión cuando su tensión arterial es demasiado elevada. De la tensión arterial se dan dos valores: i) la tensión sistólica corresponde al momento en que el corazón se contrae o late, ii) la tensión diastólica, representa la presión ejercida sobre los vasos cuando el corazón se relaja entre latidos (Organización-Mundial-Salud, 2021).

4. `Accidente cerebrovascular`: sucede cuando el flujo de sangre a una parte del cerebro

se detiene, el cerebro no puede recibir nutrientes y oxígeno. Las células cerebrales pueden morir, lo que causa daño permanente. Un accidente cerebrovascular se presenta cuando un vaso sanguíneo en el cerebro se rompe, causando un sangrado dentro de la cabeza (MedlinePlus, 2022a).

5. `Enfermedad coronaria`: se produce un estrechamiento u obstrucción de las arterias coronarias que llevan sangre y oxígeno al corazón. La causa de la cardiopatía coronaria por lo general es la ateroesclerosis, es decir, la acumulación de grasa y formación de placas dentro de las arterias coronarias. Esta enfermedad causa dolor de pecho, dificultad para respirar durante el ejercicio físico e infartos de miocardio (Instituto-Nacional-Cáncer, 2019).

Es importante aclarar que el objetivo de la investigación no estuvo centrado en la predicción de un tipo de enfermedad en particular, nos centramos en la intersección de todos los factores de riesgo y características que ayudan a diagnosticar los distintos tipos de enfermedades cardiovasculares. Esto deriva del estudio realizado a los conjuntos de datos encontrados en repositorios públicos.

2.5.1. Patrones que derivan en su diagnóstico

En nuestro país se ha reportado que el 60 % de la población adulta presenta al menos un factor de riesgo de enfermedad cardiovascular (obesidad y sobrepeso, hipertensión arterial, diabetes, tabaquismo y dislipidemias) (Cervantes, 2019). El tabaquismo se considera un factor de riesgo para las enfermedad cardiovascular, existe evidencia para llegar a la conclusión de que hubiera una relación causal entre el consumo de cigarrillos y la incidencia de enfermedad cardiovascular (Aguilar-Salinas et al., 2022). Los estudios han demostraron que los fumadores presentaban un aumento del riesgo de infarto al miocardio o muerte súbita. Actualmente los estudios indican que existe una relación directamente proporcional al número de cigarrillos consumidos al día (O'Donnela y Elosuab, 2008).

La disminución del 10 % en el colesterol sérico produce una disminución del riesgo de enfermedad cardiovascular en un 50 % a la edad de 40 años, del 40 % a los 50 años, del 30 % a los 60 años y del 20 % a los 70 años. Por otra parte, mantener concentraciones altas de colesterol

HDL (hombres $\geq$ 40 mg y mujeres $\geq$ 50 mg) proporciona una protección en la génesis de la enfermedad cardiovascular en comparación con individuos con concentraciones de colesterol HDL de 44-45 bajas (O'Donnela y Elosuab, 2008).

Birjmohun et al., 2005 mostraron que el aumento de 1 mg/dL en la concentración de colesterol HDL se asocia a una disminución del riesgo coronario de un 2 % en los varones y un 3 % en las mujeres. Por lo que actualmente la disminución en las concentraciones de colesterol HDL es factor de riesgo en el incremento de enfermedad cardiovascular y viceversa.

Los investigadores Aguilar-Salinas et al., 2022 discutieron los resultados de la Encuesta Nacional de Enfermedades Crónica (ENEC), en donde se registró en población urbana adulta de entre 20 a 69 años una prevalencia del 24.4 % de hipertrigliceridemia, el cual es un nivel elevado de un determinado tipo de grasa (triglicéridos) en la sangre. Sin embargo, la importancia que tiene el análisis que se hace de esa información radica en el hecho de la alta prevalencia de enfermedad mixta (12.8 %), personas con un promedio de edad de 42.7 años, un 56 % del sexo masculino. La concentración promedio del grupo afectado fue de colesterol (239 $\pm$ 28.23 mg/dl), triglicéridos (TG) (348.1 $\pm$ 194.86 mg/dl) y colesterol vinculado a lipoproteínas de baja densidad (c-HDL) (202.24 $\pm$ 29.39 mg/dl), significativamente mayor a las cifras esperadas. La frecuencia de comorbilidades en este grupo también se observó incrementada, diabetes (21.4 %), hipertensión arterial (42.6 %), tabaquismo (57 %), obesidad (30.7 % y sobrepeso (51.3 %).

Por su parte, la ingestión de alcohol se ha relacionado con aumento en el riesgo cardiovascular. Se considera bebedor ocasional cuando se consumen 2 o 3 sustancias elaboradas a base de alcohol por semana y bebedor frecuente cuando se consumen 4 o más sustancias elaboradas a base de alcohol por semana (J. et al., 2017). Fernández-Solá, 2005 realizó un meta-análisis de diversos estudios donde indicó que el riesgo cardiovascular se incrementa con el aumento del consumo de alcohol de más de 60g. Los accidentes cerebro vasculares isquémico, hemorrágico fueron un poco más frecuentes en comparación de 2:1 en personas que consumían dicha cantidad (riesgo similar en hombres y en mujeres). Cuando el consumo es $\geq$ 80 g al día la función ventricular izquierda se deteriora, aumenta el riesgo a desarrollo de insuficiencia cardíaca terminal o arritmias, con frecuente inducción de muerte súbita.

Capítulo 3

Trabajo relacionado

El presente capítulo aborda las aportaciones relacionadas con esta investigación, principalmente en los temas de optimización de hiperparámetros y el modelado predictivo aplicado a la predicción de enfermedades cardiovasculares. La búsqueda o creación de un enfoque efectivo para abordar este desafío ha generado numerosos estudios que exploran diversas estrategias. A continuación, se muestra el resultado del análisis de los trabajos previos con el propósito de contextualizar nuestra investigación. Los trabajos analizados se dividieron en cuatro categorías, cada una hace referencia al mecanismo de configuración de hiperparámetros aplicado, estas son: i) teoría de optimización en aprendizaje automático, ii) la configuración predeterminada, iii) configuración con base en la experiencia del investigador y iv) configuración mediante enfoque de optimización. En la Tabla 3.5 se describen a detalle cada uno de los trabajos, se listan los conjuntos de datos, los clasificadores empleados y sus correspondientes resultados.

3.1. Teoría de la optimización en aprendizaje automático

En el trabajo de Yang y Shami, 2020 se realizó una revisión de los algoritmos de aprendizaje automático tradicionales y sus hiperparámetros relevantes. Analizaron las técnicas de optimización de hiperparámetros, incluidos sus beneficios y desventajas, con el fin de ayudar a aplicarlas a diferentes modelos de aprendizaje automático. Además, se examinaron las bibliotecas y los marcos comunes de optimización para su uso práctico. Finalmente, se abordaron los desafíos abiertos y las direcciones de investigación del dominio de investigación de optimización de hiperparámetros.

Por su parte, en Andonie, 2019 presentaron una visión integrada de los métodos utilizados en la optimización de hiperparámetros de sistemas de aprendizaje, con énfasis en los aspectos de complejidad computacional. El problema que intentaron resolver incluye la combinación de técnicas de: optimización, reducción del espacio de búsqueda y tiempo de entrenamiento.

Los autores Probst et al., 2018 formalizaron el problema del ajuste desde un punto de vista estadístico, se definieron los valores predeterminados basados en datos y sugirieron medidas generales que cuantifican la capacidad de ajuste de los hiperparámetros de los algoritmos. Además, realizaron un estudio comparativo a gran escala basado en 38 conjuntos de datos de la plataforma OpenML y seis algoritmos comunes de aprendizaje automático.

Kotthoff et al., 2017 abordaron el problema de la selección simultanea de un algoritmo de aprendizaje automático y la optimización de sus hiperparámetros mediante la herramienta Auto-WEKA; en donde la técnica de optimización bayesiana, los mecanismos de selección de características y los algoritmos de aprendizaje automático implementados en el software de minería de datos WEKA fungieron como herramientas principales. Para buscar el espacio de algoritmos candidatos y sus ajustes de hiperparámetros en Auto-WEKA utilizaron el método de búsqueda estocástica, denominado *Sequential Model-Based Algorithm Configuration* (SMBO)), y una función para medir el error de clasificación.

La Tabla 3.1 muestra los artículos que consideran la configuración de hiperparámetro como problema de optimización.

ID	Año	Título	Autores
1	2020	On hyperparameter optimization of machine learning algorithms: Theory and practice	Li Yang, Abdallah Shami
2	2019	Hyperparameter optimization in learning systems	Razvan Andonie
3	2018	Tunability: Importance of Hyperparameters of Machine Learning Algorithms	Philipp Probst, Anne-Laure Boulesteix, B. Bischl
4	2017	Auto-WEKA 2.0: Automatic Model Selection and Hyperparameter Optimization in WEKA	Kotthoff Lars, Thornton Chris, Hoos H. H., Hutter Frank, Leyton-Brown Kevin

Tabla 3.1: Trabajos que abordan la configuración de hiperparámetros como problema de optimización.

3.2. Configuración predeterminada

En el trabajo de Guarneros-Nolasco et al., 2021 identificaron los atributos de riesgo cardiovascular en cuatro conjuntos de datos, enfocados principalmente a la predicción y diagnóstico de enfermedad cardiovascular. Analizaron y compararon el rendimiento de diez clasificadores para identificar los dos y cuatro atributos principales en los conjuntos de datos. Los tres factores de riesgo principales para arritmia y taquicardia detectados son: tipo de dolor en el pecho, nivel de creatinina en sangre, y porcentaje de sangre que sale del corazón en cada latido. Los clasificadores que exhibieron mayor porcentaje en la métrica *accuracy* usando k-folds CV fueron LR y SVM en los conjuntos de datos de Cleveland y Framingham, con dos y cuatro atributos principales. Para el conjunto de datos de Faisalabab, fue el modelo DT con dos atributos, y el modelo CatBoost considerando los cuatro atributos principales.

Reddy et al., 2021 emplearon dos enfoques de selección de características para entrenar diez clasificadores. En el primer enfoque se utilizó el conjunto de datos completo. En el segundo enfoque, se utilizaron las técnicas de selección de atributos: *Correlation-Based Feature, Chi-Squared y ReliefF*. El proceso experimental determinó que el modelo *Sequential Minimal Optimizatio* (SMO) alcanzó el mejor desempeño en la métrica *accuracy* en el primer enfoque. Para el segundo enfoque también obtuvo el mejor desempeño cuando se entrenó con el subconjunto de atributos obtenido por la técnica *chi-squared*. Por su parte, el clasificador *bagging* con LR proporcionó el área bajo la cuerva ROC más alta, utilizando los conjuntos de atributos completos y el subconjunto seleccionado por la técnica *ReliefF*.

Shah et al., 2020 presentaron varios atributos del conjunto de datos Cleveland relacionados con las enfermedades del corazón, para realizar el entrenamiento de los clasificadores *Naive Bayes*, DT, *K-Nearest Neighbor* (k-NN) y RF en el software de minería de datos WEKA. Este trabajo describió además qué atributos contribuyen más a la previsión de una mayor exactitud de predicción de enfermedad cardiovascular. En la fase de preprocesamiento del conjunto de datos se realizó la limpieza, transformación, integración y reducción de los datos. Tras aplicar los algoritmos, el porcentaje de *accuracy* más alto se obtuvo con el algoritmo de K-NN.

Uddin et al., 2019 identificaron las tendencias clave entre los diferentes tipos de algoritmos de aprendizaje automático. Además, se resumieron sus ventajas y limitaciones. El hecho de considerar de sólo los artículos que utilizaron datos clínicos y demográficos (15 artículos) revela que el modelo DT muestra el resultado superior en la mayoría de las ocasiones. La SVM mostró un *accuracy* superior en la mayoría de las ocasiones para tres enfermedades (por ejemplo, enfermedades cardíacas, diabetes y enfermedad de Parkinson). De los 17 estudios en los que se aplicó, RF mostró la mayor exactitud en 9 de ellos, es decir, el 53 %. Le siguió SVM, que alcanzó el máximo en el 41 % de los estudios en los que se tuvo en cuenta.

La Tabla 3.2 muestra los artículos que realizan el modelado considerando el valor por defecto de los hiperparámetros.

ID	Año	Título	Autores
1	2021	Identifying the Main Risk Factors for Cardiovascular Diseases Prediction Using Machine Learning Algorithms	Guarneros-Nolasco Luis Rolando, Cruz-Ramos Nancy Aracely, Alor-Hernández Giner, Rodríguez-Mazahua Lisbeth, Sánchez-Cervantes José Luis
2	2021	Heart Disease Risk Prediction Using Machine Learning Classifiers with Attribute Evaluators	Karna Vishnu Vardhana Reddy, Irraivan Elamvazuthi, Azrina Abd Aziz, Sivajothi Paramasivam, Hui Na Chua, S. Pranavanand
3	2020	Heart Disease Prediction using Machine Learning Techniques	Devansh Shah, Samir Patel, Santosh Kumar Bharti
4	2019	Comparing different supervised machine learning algorithms for disease prediction	Shahadat Uddin, Arif Khan, Md Ekramul Hossain, Mohammad Ali Moni

Tabla 3.2: Trabajos que abordan la configuración predeterminada de los hiperparámetros.

3.3. Configuración con base en la experiencia del usuario

Allah et al., 2022 realizaron un análisis comparativo de cinco clasificadores para llegar a la decisión más adecuada para diagnosticar enfermedad cardiovascular con mayor exactitud que los modelos existentes. Esto se realizó en dos conjuntos de datos diferentes de enfermedades cardíacas. Además, en los experimentos fueron seleccionados los valores de los hiperparámetros mediante prueba y error. Los modelos RF, SVM y XGBoost obtuvieron mejor desempeño debido a su capacidad de predicción superior a la de otros modelos, con un *accuracy* mayor al 91,6 % en los dos conjuntos de datos.

El marco de trabajo propuesto por Gupta et al., 2020 se analizaron cinco clasificadores entrenados con la combinación óptima del conjunto de atributos y un respectivo ajuste de los hiperparámetros con base en la experiencia de los investigadores. Dado que el conjunto de datos es de tipo mixto y contiene características tanto numéricas como categóricas, implementaron la técnica *Factor Analysis of Mixed Data* (FAMD) para la extracción de características. Además, aplicaron los métodos: codificación *One-Hot* para las características categóricas y *Z-core Normalization* para la estandarización de los datos. El entrenamiento mostró que el modelo RF muestra el mayor rendimiento en la predicción de enfermedades cardiovasculares con las métricas *accuracy, recall* y *specifity*.

Ayon et al., 2020 realizaron un estudio comparativo de siete clasificadores para la predicción de la enfermedad de las arterias coronarias. Se realizó la configuración de los hiperparámetros de todos los algoritmos con base a la experiencia de los investigadores. El rendimiento de cada técnica se evaluó utilizaron dos conjuntos de datos de enfermedades cardiovasculares del repositorio de aprendizaje automático de la UCI. En el preprocesamiento de los dos conjuntos de datos se corrigieron los datos faltantes y datos atípicos. Como resultado del entrenamiento se determinó que la Red Neuronal Profunda y el modelo SVM obtuvieron mayor capacidad predictiva en términos de la métrica *accuracy*.

Li et al., 2020 propusieron un nuevo enfoque de selección de características llamado *Fast Conditional Mutual Information* (FCMIM). Además, aplicaron los mecanismos *Relief, Minimal Redundancy Maximal Relevance, LASSO y Local Learning* con el objetivo de realizar un análisis comparativo del desempeño de los modelos de aprendizaje automático y determinar si el

enfoque propuesto logra reducir el tiempo de ejecución. En la fase de preprocesamiento se eliminaron los registros con datos perdidos y se creó una escala unitaria para todos los atributos mediante las técnicas *Standard Scalar, Min-Max Scalar*. Además, se utilizó el método *Leave One Subject Out* para la evaluación de los modelos y para el ajuste de hiperparámetros. El sistema FCMIM con base en el modelo SVM alcanzó una buen *accuracy* en comparación con los métodos convencionales.

La Tabla 3.3 muestra los artículos que realizan modelado y configuración de los hiperparámetros basados en el conocimiento del usuario.

ID	Año	Título	Autores
1	2022	Performance Comparison of Various Machine Learning Approaches to Identify the Best One in Predicting Heart Disease	Enas M. Abd Allah, Doaa E. El-Matary, Esraa M. Eid, Adly S. Tag El Dien
2	2020	MIFH: A Machine Intelligence Framework for Heart Disease Diagnosis	Gupta Ankur, Kumar Rahul, Singh Arora Harkirat, Raman Balasubramanian
3	2020	Coronary Artery Heart Disease Prediction: A Comparative Study of Computational Intelligence Techniques	Safial Islam Ayon, Md. Milon Islam, Md. Rahat Hossain
4	2020	Heart Disease Identification Method Using Machine Learning Classification in E-Healthcare	Li Jian Ping, Haq Amin Ul, Din Salah Ud, Khan, Jalaluddin, Khan Asif, Saboor Abdus

Tabla 3.3: el modelad que realizan el modelado aplicando una configuración manual de los hiperparámetros.

3.4. Configuración mediante enfoques de optimización

En el trabajo de Asadi et al., 2021 se aplicaron los enfoques *Multi-objective Particle Swarm Optimization* para entrenar el algoritmo RF y lograr una predicción de enfermedades cardiovasculares eficiente. El objetivo principal fue producir árboles de decisión diversos y precisos y determinar el número óptimo de ellos simultáneamente. El método utiliza un enfoque evolutivo multiobjetivo, es decir, selección de características en RF y selección aleatoria del conjunto de entrenamiento.

En Rohit et al., 2021 aplicaron el enfoque *Grid Search* pata optimizar los hiperparámetros de seis clasificadores y un modelo de aprendizaje profundo. En el análisis de los conjuntos de datos se detectaron valores atípicos y la distribución de los datos. Por ello utilizaron tres enfoques, en el primero excluyeron el tratamiento de los valores atípicos y el proceso de selección de atributos;

en el segundo enfoque solo realizaron la selección de atributos; en tercer enfoque, el conjunto de datos fue normalizado para la detectar valores atípicos y para seleccionar los atributos. De igual forma, en el trabajo de Ghosh et al., 2021 aplicaron el mismo enfoque de optimización de hiperparámetros para entrenar clasificadores híbridos, los cuales se desarrollan integrando los clasificadores tradicionales con los métodos *bagging* y *boosting*. Además, las técnicas de selección de atributos Relief y LASSO ayudaron a extraer las características más relevantes. Con ello, el modelo híbrido RF presentó mayor rendimiento con la técnica Relief y para el caso de la técnica LASSO, el modelo híbrido *Gradient Boosting*.

Valarmathi y Sheela, 2021 trataron el problema del desbalance de clases con la técnica *Synthetic Minority Over-sampling Technique* (SMOTE). Eliminaron las características irrelevantes con la técnica *Sequential Forward Selection*. En el análisis comparativo de los dos modelos se realizó la optimización de los hiperparámetros mediante los enfoques *Grid Search, Random Search y Tree-based Pipeline Optimization Tool* (TPOT). Como resultado, el modelo RF optimizado con TPOT obtuvo el mejor desempeño en la métrica *accuracy*, en el conjunto de datos Cleveland. Para el caso del conjunto de datos Z-Alizadeh Sani, RF con el enfoque *Random Search* mostró mayor *accuracy* para el diagnóstico de estenosis de las arterias.

Kabir y Zaman, 2020 presentaron dos enfoques de diseño de modelos de predicción. El enfoque tradicional se realizó sin ningún método de ajuste de hiperparámetros, es decir, se utilizan los parámetros por defecto para generar los modelos de clasificación. En el segundo enfoque se propuso la aplicación de la técnica de optimización *Grid Search* mediante un proceso de validación cruzada para ajustar los hiperparámetros. En la fase final, ambos enfoques evaluaron el rendimiento del conjunto de entrenamiento y del conjunto de prueba. Finalmente, se identificaron mejores resultados con los sistemas de predicción con ajuste de hiperparámetros en todos los modelos entrenados.

Budholiya et al., 2022 establecieron un sistema de predicción de enfermedad cardiovascular con base en el algoritmo XGBoost. Para obtener la solución óptima de forma sistemática y eficaz aplicaron el enfoque de optimización bayesiana, para determinar los valores óptimos de los hiperparámetros. Los resultados del modelado con XGBoost se compararon con los modelos RF y *Extra Tree* (ET). Estos dos clasificadores también fueron entrenados con el mismo conjunto de datos y se realizó su correspondiente optimización de hiperparámetros. Como resultado, el

modelo XGBoost mostró mejor rendimiento en comparación con los dos modelos RF y ET.

La Tabla 3.4 muestra los artículos que aplicaron enfoques de optimización de hiperparámetros.

ID	Año	Título	Autores	Enfoque
1	2021	Random forest swarm optimization-based for heart diseases diagnosis	Shahrokh Asadi, SeyedEhsan Roshan, Michael W. Kattan	Multi-objective Particle Swarm Optimization
2	2021	Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning	Rohit Bharti, Aditya Khamparia, Mohammad Shabaz, Gaurav Dhiman, Sagar Pande, Parneet Singh	Grid Search
3	2021	Efficient Prediction of Cardiovascular Disease Using Machine Learning Algorithms With Relief and LASSO Feature Selection Techniques	Ghosh Pronab, Azam Sami, Jonkman Mirjam, Karim Asif, Shamrat F. M. Javed Mehedi, Ignatious Eva, Shultana Shahana, Beeravolu Abhijith Reddy, De Boer Friso	Grid Search
4	2021	Heart disease prediction using hyper parameter optimization (HPO) tuning	R. Valarmathi, T. Sheela	Grid Search, Random Search, TPOT
5	2020	Developing a Hyperparameter Tuning Based Machine Learning Approach of Heart Disease Prediction	Hashi Emrana Kabir, Md. Shahid Uz Zaman	Grid Search
6	2022	An optimized XGBoost based diagnostic system for effective prediction of heart disease	Kartik Budholiya, Shailendra Kumar Shrivastava, Vivek Sharma	Bayesian optimization

Tabla 3.4: Trabajos que aplican enfoques de optimización de hiperparámetros.

3.5. Conclusiones del estado del arte

Se exploraron diversas propuestas y estrategias para el proceso de modelado y configuración de hiperparámetros. A pesar de la disponibilidad de configuraciones predefinidas, es evidente que los expertos siguen probando con diferentes combinaciones de hiperparámetros, ya sea basándose en la experiencia o mediante un enfoque de prueba y error. Aunque existe una variedad de enfoques de optimización en el estado del arte, se observó en la revisión que muchos de los que se aplicaron son enfoques exhaustivos y requieren una gran cantidad de tiempo, lo que plantea desafíos adicionales en la búsqueda de configuraciones óptimas (ver Tabla 3.5). Estos hallazgos dejan claro la importancia continua de investigar y aplicar otros enfoques eficientes para abordar la configuración de hiperparámetros como un problema de optimización.

Referencia	Descripción del trabajo	Conjunto de datos	Mejor algoritmo de ML	Desempeño del modelo
Guarneros-Nolasco et al., 2021	Identificación de dos y cuatro mejores atributos de los conjuntos de datos de enfermedad cardiovascular analizando el rendimiento de la métrica *accuracy* para determinar que son los mejores para predecir y diagnosticar la enfermedad cardiovascular.	Cleveland, Framingham, Faisalabad, South African Hearth	RF, LR, SVM, XGBoost	2-Atributos % Acc: 81.32 (Cleveland); 77.84 (Framingham); 74.44 (Faisalabad); 72.38 (South African Hearth); 4-Atributos % Acc: 82.42 (Cleveland); 81.18 (Framingham); 71.11 (Faisalabad); 71.22 (South African Hearth)
Reddy et al., 2021	Modelado de diez clasificadores de diferentes categorías utilizando el conjunto completo de atributos y los subconjuntos de atributos de tres técnicas evaluadoras de atributos	Cleveland	SMO, Bagging-LR	% Acc: 85.14, 84.48; % spe: 90, 88 (14 atributos); % Acc: 83.82, 83.82; % Spe: 88, 87 (C-BF); % Acc: 86.46, 85.47; % Spe: 90, 89 (chi-squared); % Acc: 86.13, 85.14; % Spe: 90, 88 (ReliefF)
Shah et al., 2020	Prever la probabilidad de desarrollar una cardiopatía en los pacientes a partir de algoritmos de aprendizaje de máquina	Cleveland	K-NN, Naive Bayes, RF	% Acc: 90.78, 88.15, 86.84

Referencia	Descripción del trabajo	Conjunto de datos	Mejor algoritmo de ML	Desempeño del modelo
Uddin et al., 2019	Revisión de estudios que aplican algoritmos de aprendizaje automático para la predicción de enfermedad cardiovascular con el objetivo de identificar aquellos algoritmos de mayor desempeño	Scopus, PubMed	RF, SVM	SVM se aplica con mayor frecuencia. Sin embargo, el algoritmo Random Forest (RF) mostró mayor desempeño en términos de la métrica *accuracy*.
Allah et al., 2022	Análisis comparativo de seis modelos de aprendizaje automático para llegar a la decisión que mejor apoye el diagnóstico de las cardiopatías con mejor exactitud que los modelos existentes	Cleveland, Kaggle	RF, SVM, XGBoost	% Acc: 87.5, 90.9, 91.6 (Cleveland), 100, 100, 100 (Kaggle); % Prec: 80, 80, 87 (Cleveland), 99.5, 98, 99.5 (Kaggle); % Sen: 80, 80, 88 (Cleveland), 99.5, 98, 99.5 (Kaggle)
Gupta et al., 2020	Análisis Factorial de Datos Mixtos para extraer y derivar características del conjunto de datos sobre enfermedad cardiovascular. Además, ajuste manual de los hiperparámetros para entrenar los modelos.	Cleveland	RF, SVM, LR	% Acc: 91.80, 91.80, 93.44; % Spe: 90.90, 84.84, 96.96; % Sen: 92.85, 100, 89.28

Referencia	Descripción del trabajo	Conjunto de datos	Mejor algoritmo de ML	Desempeño del modelo
Ayon et al., 2020	Entrenamiento de siete técnicas de inteligencia computacional para la predicción de cardiopatías coronaria con datos del repositorio de aprendizaje automático de la UCI.	Statlog, Cleveland	Statlog: DNN; Cleveland: SVM	% Acc: 98.15, 97.36; % Sen: 98.67, 98.78; % Spe: 97.50, 95.68; % Prec: 98.01, 96.43
Li et al., 2020	Selección de características con Relief, Minimal-Redundancy-Maximal Relevance, Lasso, Local Learning Based Feacture Selection y entrenamiento de algoritmos con diferentes configuraciones de hiperparámetros	Cleveland	SVM	% Acc: 92.37; % Sen: 89.00; % Spe: 98.00
Asadi et al., 2021	Aplicar Optimización de Enjambre de Partículas Multiobjetivo para determinar la mejor combinación de hiperparámetros del algoritmo RF	Statlog, Cleveland, SPECT, SPECTF, VA, Eric	RF	% Acc: 88.26, 85.21, 87.65, 86.70, 87.50, 80.95

Referencia	Descripción del trabajo	Conjunto de datos	Mejor algoritmo de ML	Desempeño del modelo
Rohit et al., 2021	Combinación de algoritmos de inteligencia artificial mediante tres enfoques y la configuración de hiperparámetros a través de un enfoque exhaustivo	Cleveland, Hungary, Switzerland, and Long Beach V.	kNN, ANN	% Acc: 84.8, 94.2; % Spe: 77.7, 83.1; % Sen: 85, 82.3
Ghosh et al., 2021	Se entrenam algoritmos híbridos a partir de Selección de atributos relevantes con Relief y LASSO en función de los valores de clasificación en las referencias médicas	Cleveland, Long Beach VA, Switzerland, Hungarian and Stat log	RF Bagging, Gradient Boosting	% Acc RFB: 99.05 Relief, 97.65 LASSO; % Acc GBB: 98.32 Relief, 97.85 LASSO
Valarmathi y Sheela, 2021	Se aplicaron los enfoques Grid Search, Random Search, Tree-based Pipeline Optimization Tool para optimizar el rendimiento de RF y XGBoost	Cleveland, Z-Alizadeh Sani	RF, XGBoost	% Acc RF: 91.32, 95.04, 97.52; % Acc XGBoost: 86.36, 92.14, 90.50; % Spe RF: 93.89, 96.95, 97.70; % Spe XGBoost: 89.31, 93.89, 93.12

Referencia	Descripción del trabajo	Conjunto de datos	Mejor algoritmo de ML	Desempeño del modelo
Kabir y Zaman, 2020	Sistema de predicción en el que se entrenan cinco algoritmos con el enfoque Grid Search para la optimizacón de hiperparámetros.	Cleveland	LR, kNN, SVM	% Acc: 90.16, 91.80, 90.16; % Sen: 87.50, 90.62, 87.50; % Prec: 93.33, 93.55, 93.33
Budholiya et al., 2022	Sistema eficiente de diagnóstico para enfermedades cardiovasculares mediante la optimización de los hiperparámetros de los algoritmos XGBoost, RF y Extra-Trees	Cleveland	XGBoost	% Acc: 91.80; % Spe: 96.96; % Sen: 85.71; % F1-Sco: 90.56; % ROC-AUC: 91.34

Tabla 3.5: Comparación de los trabajos relacionados.

Capítulo 4

Optimización de hiperparámetros

Este capítulo describe las actividades que conforman la metodología de solución del proyecto de investigación, dichas actividades se mostraron en la Figura 1.1 del Capítulo 1. Se presentan los resultados del análisis exploratorio y el preprocesamiento de los conjuntos de datos Cleveland, Framingham y Faisalabad. Estos conjuntos de datos públicos almacenan información del dominio de las enfermedades cardiovasculares, en particular, factores de riesgo, características fisiológicas y demográficas que ayudan en el diagnostico de paciente con riesgo cardiovascular. Además, se describen las pruebas experimentales del proceso de optimización mediante los cinco enfoques de optimización, esto incluye la propuesta de los espacios de búsqueda para cada algoritmo de aprendizaje automático así la inicialización de los parámetros de los enfoques de optimización PSO y GA.

4.1. Etapas del proceso de optimización

El proceso de optimización de hiperparámetros aplicado en esta investigación se conforma de distintas fases. Iniciando con el análisis exploratorio y el preprocesamiento de datos. Posteriormente el entrenamiento de los modelos (proceso de optimización de los hiperparámetros) y evaluación de los modelos. Finalmente, el análisis de las configuraciones de hiperparámetros relevantes. En la Figura 4.1 se muestran el diagrama del proceso de optimización de hiperparámetros propuesto.

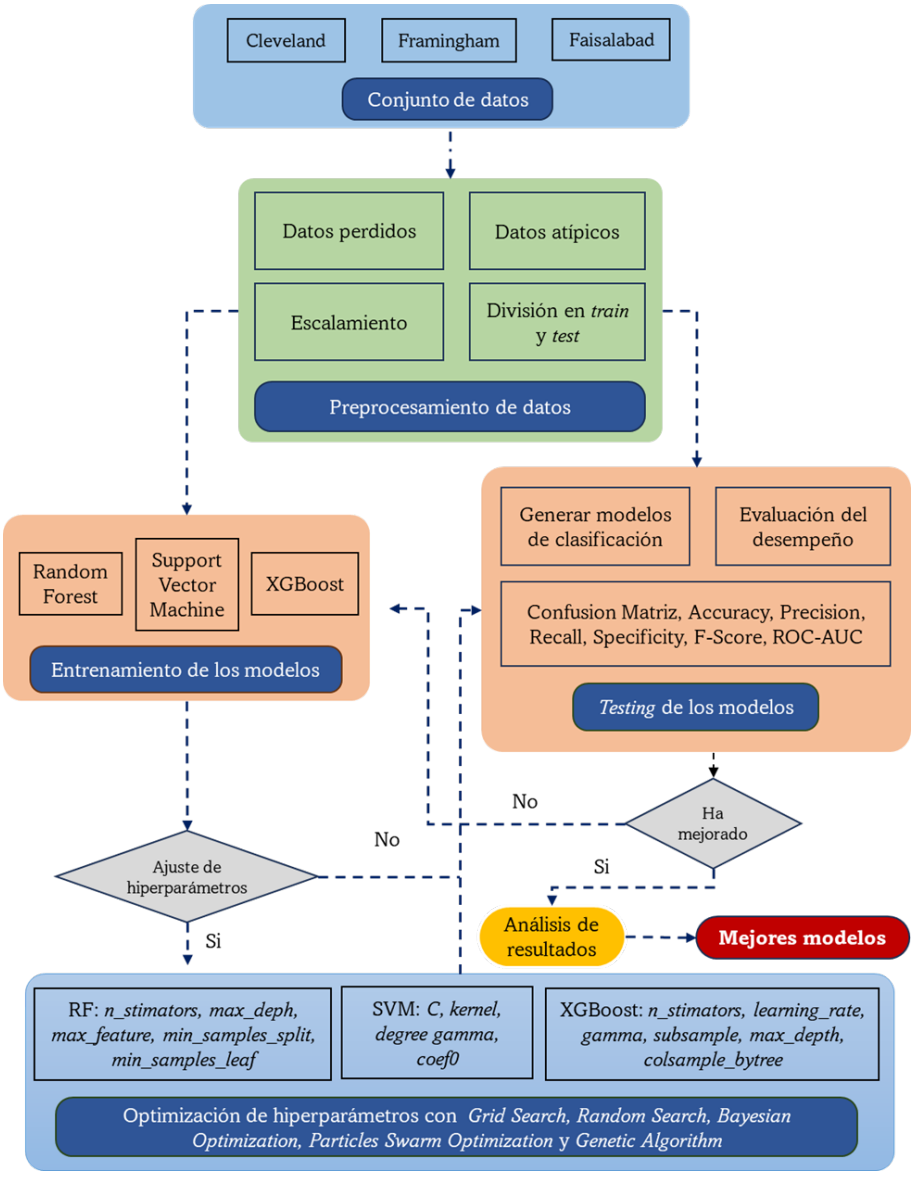


Figura 4.1: Proceso de Optimización de hiperparámetros aplicado en esta investigación.

4.2. Conjuntos de datos procesados

Comenzando con la descripción de las etapas del proceso de optimización de hiperparámetros mencionadas en la figura anterior, a continuación, se describen los conjuntos de datos utilizados para el entrenamiento de los algoritmos.

- `Cleveland`: fue recopilado por el Instituto de Cardiología de Cleveland en la década de 1980 con objetivo de predecir la presencia de enfermedades cardiovasculares en los pacientes. Se encuentra disponible en el repositorio de la Universidad de California, Invirne (UCI, 2019). Cleveland contiene información demográfica, antecedentes médicos, signos vitales de 303 personas a los que se le aplicaron una serie de pruebas de diagnóstico; de ellos, 164 presentan enfermedad cardiovascular, los 139 restantes no presentan dicha enfermedad.

- `Framingham` (National-Heart y Blood-Institute, 2022): es un estudio que se estableció originalmente en 1948 bajo el Servicio de Salud Pública de EE.UU. y posteriormente se transfirió al Instituto Nacional del Corazón, NIH, en 1949. En este estudio se reclutaron a 5.209 hombres y mujeres de entre 28 y 62 años de Framingham, Massachusetts (Abohelwa et al., 2023). Framingham contiene 4240 registros y 15 atributos. La variable objetivo ayuda a determinar si un paciente tiene una probabilidad a 10 años de desarrollar enfermedad coronaria.

- `Faisalabad` (Institute-Cardiology y hospital Faisalabad-Pakistan, 2017): corresponde a un estudio que se centró en el análisis de la supervivencia de los pacientes con insuficiencia cardíaca ingresados en el Instituto de Cardiología y el hospital aliado de Faisalabad (Pakistán) entre abril y diciembre de 2015. Faisalabad contiene 299 registros de pacientes con insuficiencia cardíaca que fueron recopilados durante el periodo de seguimiento. 194 pacientes de sexo masculino y 105 de sexo femenino con edades comprendidas entre los 40 y los 95 años.

4.3. Análisis exploratorio y preprocesamiento de datos

El objetivo de esta fase consistió en extraer información, identificar patrones y obtener conocimiento de los conjuntos de datos disponibles. El análisis exploratorio de los conjuntos de datos Cleveland, Framingham y Faisalabad se realizó con el objetivo de identificar la distribución de las características, detectar los datos tanto faltantes como atípicos. Por otro lado, se realizó el preprocesamiento de los datos con la finalidad de darle formato los datos y garantizar la calidad del análisis por parte de los modelos de aprendizaje automático.

4.1.1 Conjunto de datos Cleveland

En la Tabla 4.1 se describen los 13 atributos y la variable objetivo que contiene el conjunto de datos Cleveland.

Variables	**Descripción**	**Rango**
Age	Edad del paciente en años	29-77
Sex	Sexo del paciente	0: hombre, 1: mujer
Cp	Dolor torácico	1: AT, 2: AA, 3: DNA, 4: A
Trestbps	Tensión arterial en reposo	94-200
Chol	Colesterol sérico	126-564
Fbs	Azúcar en sangre en ayunas >120 mg/dl	0: No, 1: Si
Restecg	Resultados del electrocardiograma	0: normal, 1: anomalía-onda ST-T, 2: hipertrofia del LV
Thalach	Frecuencia cardíaca alcanzada	71-202
Exang	Angina inducida por el ejercicio	0: No, 1: Si
Oldpeak	Depresión del segmento ST	0.0-62.0
Slope	Pendiente del segmento ST de ejercicio máximo	1: pendiente-ascendente, 2: grasa, 3: pendiente-descendente
Ca	Número de vasos mayores coloreados por fuoroscopia	0-3
Thalassemia	Trastorno sanguíneo	3: normal, 6: defecto-fijo, 7: defecto irrlevante.
HeartDisease	Variable objetivo	0: ausencia d, 1: presencia

Tabla 4.1: Descripción de las variables del conjunto de datos Cleveland.

En la Tabla 4.3 se describen los estadísticos descriptivos de las variables cuantitativas de los

conjuntos de datos Cleveland. Podemos observar que el conjunto de datos contiene registros de personadas jóvenes adultas, mayores y de la tercera edad (rango 29-77 años). En relación con el sexo de los pacientes, el 32 % son mujeres y 68 % hombres.

Variables	Min	Max	Media	Mediana	Moda	Desv. Est.	Varianza
age	29	77	54.43	56	58	9.039	81.69
trestbps	94	200	131.68	130	120	17.60	309.75
chol	126	564	246.69	241	197	51.77	2680.84
thalach	71	202	149.60	153	162	22.87	523.26
oldpeak	0	6.2	1.039	0.80	0	1.161	1.34

Tabla 4.3: Estadísticos descriptivos de las variables cuantitativas del conjunto de datos Cleveland.

4.1.1.1 Análisis de correlación

La correlación entre variables es una parte importante en la fase de preprocesamiento de los datos. Ayuda a identificar cualquier relación importante que permita comprender mejor los datos, además, esto puede ayudar a la identificar las variables redundantes (Allah et al., 2022).

Existen distintas técnicas para medir la correlación, para este conjunto de datos aplicamos la correlación de *Spearman* ya que es menos sensible a los valores atípicos y es más adecuada para variables que no presentan normalidad en su distribución (El-Hashash y Shiekh, 2022).

La Figura 4.2 muestra la matriz de correlación de las características cuantitativas del conjunto de datos Cleveland. Las tuplas de variables que presentan mayor correlación son: [depresión del segmento ST - Frecuencia cardíaca, Frecuencia cardíaca - edad y Tensión arterial – edad]. Las dos primeras correlaciones son negativas débiles, lo que indica que a medida que una variable aumenta, la otra variable disminuye. Por otro lado, la última relación es positiva débil, lo que nos hace inferir que a medida que aumenta la edad en las personas también su tensión arterial incrementa.

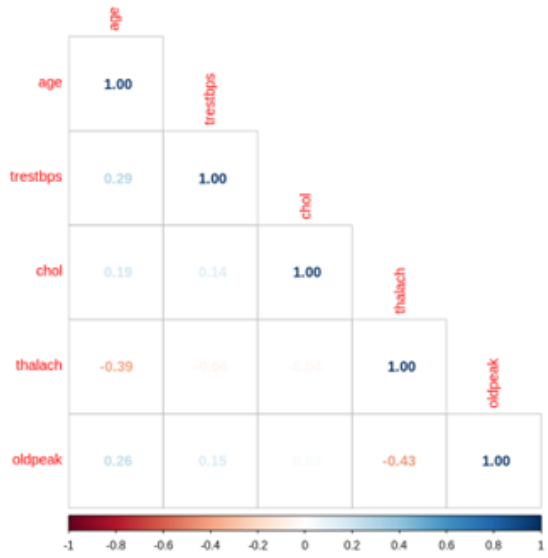
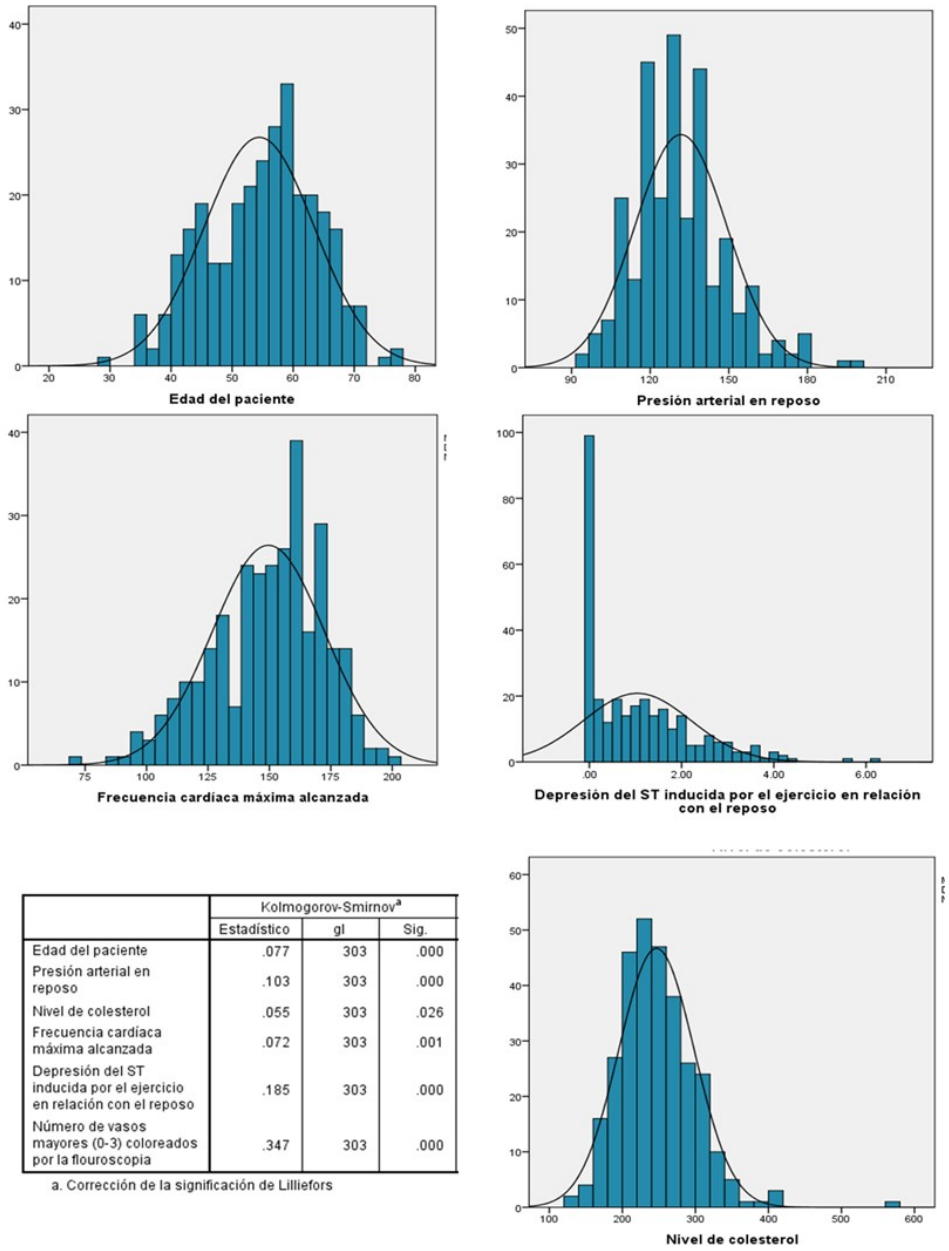


Figura 4.2: Matriz de correlación de las variables cuantitativas del conjunto de datos Cleveland.

4.1.1.2 Prueba de normalidad

La Figura 4.3 muestra las distribuciones de las características cuantitativas mediante los métodos: i) histogramas de frecuencia y ii) prueba *Kolmogorov Smirnov* (KS). Se observa en los histogramas de frecuencia que las frecuencias de los valores en algunas características sobrepasan la cuerva de normalidad de referencia. Además, existen atributos que tienen sesgos en su distribución, tal es el caso de los atributos *oldpeak, chol.*



	Kolmogorov-Smirnov[a]		
	Estadístico	gl	Sig.
Edad del paciente	.077	303	.000
Presión arterial en reposo	.103	303	.000
Nivel de colesterol	.055	303	.026
Frecuencia cardíaca máxima alcanzada	.072	303	.001
Depresión del ST inducida por el ejercicio en relación con el reposo	.185	303	.000
Número de vasos mayores (0-3) coloreados por la flouroscopia	.347	303	.000

a. Corrección de la significación de Lilliefors

Figura 4.3: Histogramas de frecuencia y prueba *Kolmogorov Smirnov* de las variables cuantitativas del conjunto de datos Cleveland.

Por otro lado, en la tabla de la misma figura se muestra el resultado de la prueba KS, donde observamos que los valores de cada renglón (prueba KS por atributo) de la columna Sig. son menores al umbral 0.05, esto nos indica que no existe evidencia estadística para aceptar la hipótesis nula, es decir, no existe normalidad en las características. En general, con la prueba KS resultó que los cinco atributos no siguen una distribución normal.

4.1.1.3 Identificación de datos atípicos

Un tema muy importante en el análisis exploratorio de los datos es poder identificar los datos que se encuentran fuera del rango normal o esperado de los datos, llamados datos atípicos o outliers. Se empleó el mecanismo de cajas y bigotes para observar analíticamente aquellas características que contienen estos datos. En el diagrama de cajas y bigotes de la Figura 4.4 se observa que las características *trestbps, chol, thalach* y *oldpeak* contienen puntos fuera de los extremos (bigotes), estos puntos hacen referencia a datos atípicos.

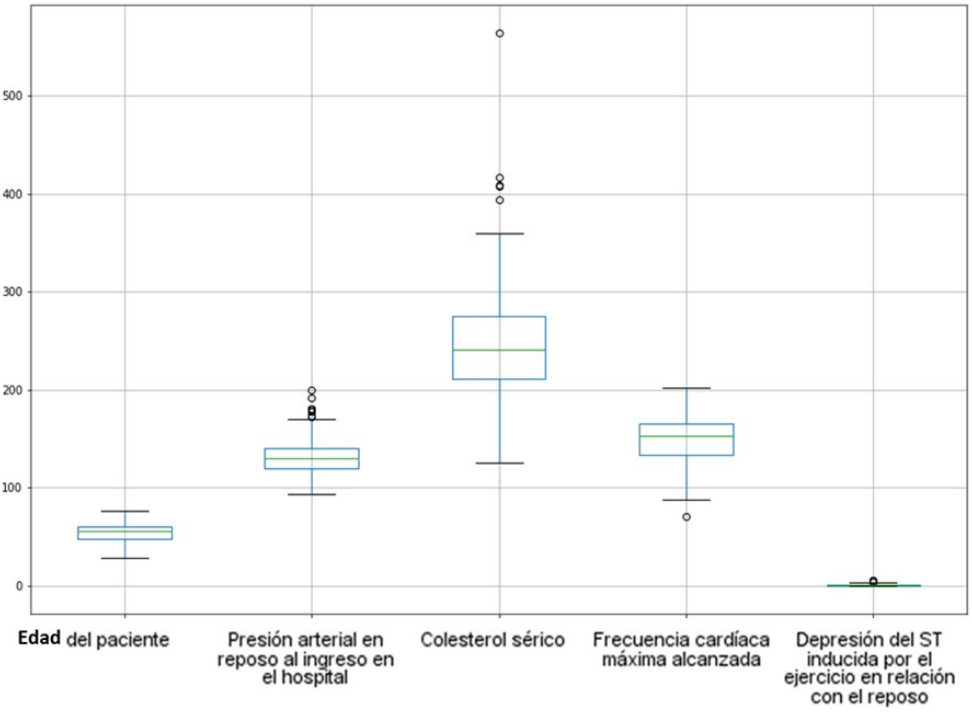


Figura 4.4: Variables cuantitativas de conjunto de datos Cleveland.

4.1.1.4 Preprocesamiento de datos

Para el tratamiento de los datos atípicos se consideró un mecanismo que los reemplaza por los valores más cercanos dentro de un rango definido. Para este caso, en cada atributo se consideró como *outliers* el 1 % como extremo inferior y el 3 % extremo superior de la distribución. La Figura 4.5 muestra las cajas y bigotes de las características con los datos atípicos tratados mediante la técnica Winsorización.

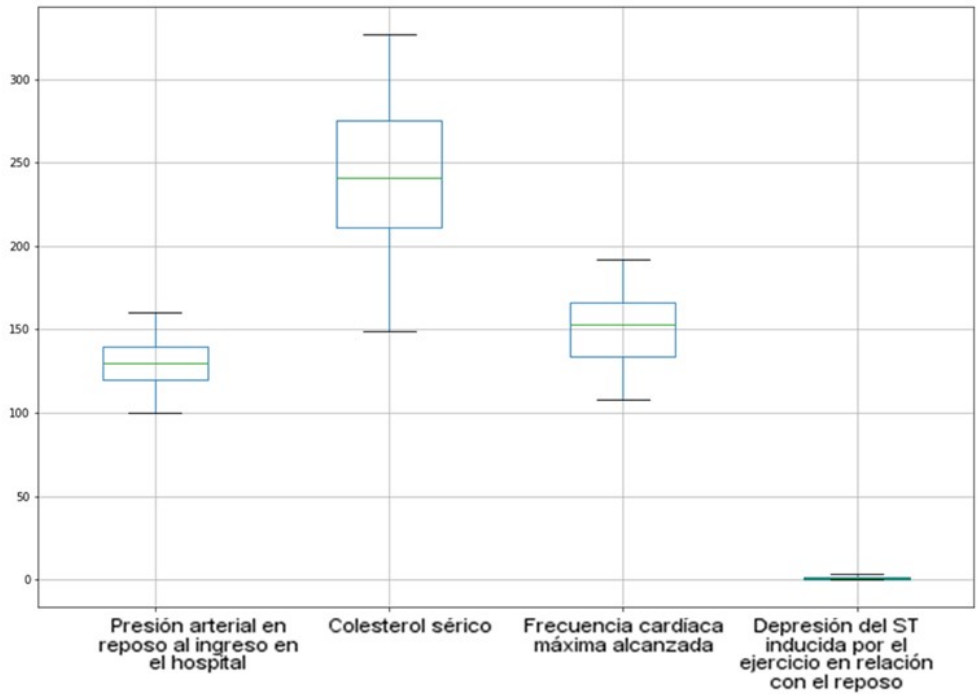


Figura 4.5: Variables cuantitativas del conjunto de datos Cleveland sin datos atípicos.

Numerosos algoritmos de aprendizaje automático requieren que las variables de entrada y salida se representen numéricamente, ya que no pueden procesar directamente datos de etiquetas en su forma original. El método *OneHot Encoding* implica la creación de una columna adicional para cada valor único presente en la variable categórica original. Cada columna se asocia con un valor binario de 0 o 1, indicando la presencia o ausencia del registro en esa categoría específica (Budholiya et al., 2022).

El conjunto de datos Cleveland presenta variables categóricas las cuales fueron transformadas a partir de este método. Las variables que se procesaron son: *cp, restecg, slope, ca y thalassemia*. Además, cuando los atributos tienen distintas escalas, es importante tener en cuenta que esto puede afectar la interpretación de los análisis y los modelos estadísticos. Es posible que las variables con mayores escalas tengan una influencia desproporcionada en los resultados en comparación con las variables con escalas más pequeñas. Se ha identificado que los datos de dominios médicos son en su mayoría discretos, por lo que estandarizarlos es esencial para hacer converger las características de estos. Para este conjunto de datos se consideró el mecanismo *Robust Scale*. Es uno de los métodos más populares para definir una única escala

para todas las características (ver diagrama de cajas y bigotes en la Figura 4.6) utilizando los percentiles 25 y 75 de la distribución de datos para cada atributo.

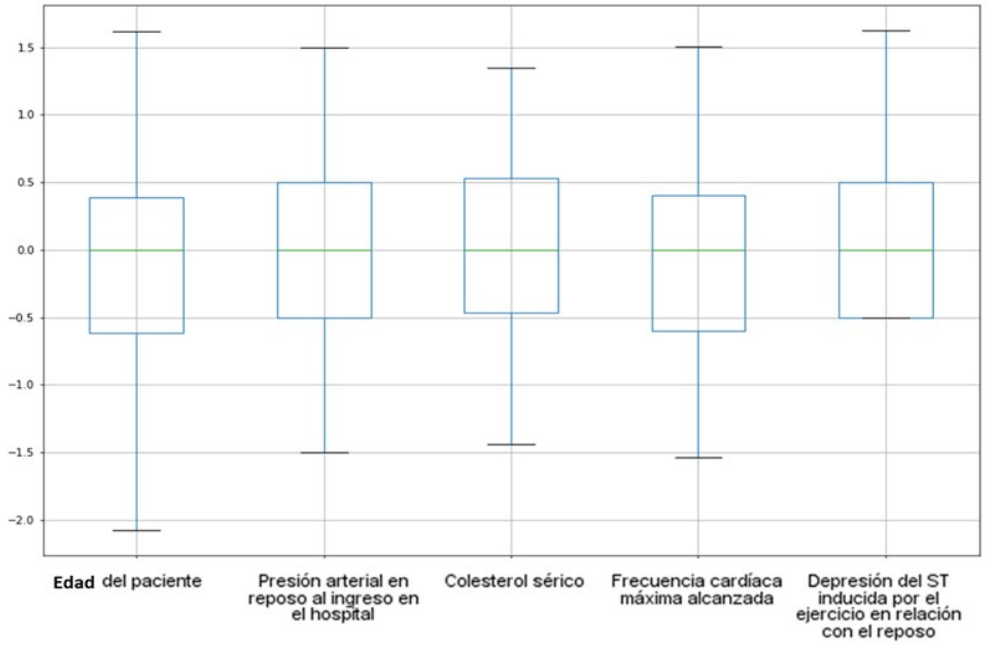


Figura 4.6: Variables cuantitativas del conjunto de datos Cleveland escaladas.

4.1.1.5 Selección de características

La selección de características en conjuntos de datos de dominios médicos, como las enfermedades cardiovasculares, es un proceso fundamental para identificar las variables más relevantes y significativas en relación con la condición de interés. Esto ayuda a reducir la dimensionalidad de los datos y mejorar la precisión de los modelos predictivos o de clasificación utilizados en el análisis. En la Tabla 4.4 se detalla el método empleado para seleccionar las características del conjunto de datos Cleveland

Método	**Descripción**	**Característica**
Information Gain	La ganancia de información mide cuánta información se gana acerca de la variable objetivo al conocer el valor de una característica particular. En otras palabras, evalúa cuánto reduce la incertidumbre en la predicción de la variable objetivo al considerar una característica en particular.	*cp, chol, thalach, exang, oldpeak, slope, ca, thal*

Tabla 4.4: Técnica de selección de atributos aplicada al conjunto de datos Cleveland.

4.1.2 Conjunto de datos Framingham

Framingham es un conjunto de datos que contiene 15 características, entre ellas información sobre factores de riesgo, información demográfica, antecedentes médicos, así como información sobre la incidencia de enfermedad cardiovascular (ver Tabla 4.5). La proporción de registros con respecto al sexo de los pacientes es de 57 % para las mujeres y 43 % para los hombres.

Variables	**Descripción**	**Rango**
Male	Sexo del paciente	0: hombre, 1: mujer
Age	Edad del paciente	32-70
Education	Nivel educativo alcanzado	1: SI, 2: SC, 3: ET, 4: U
CurrentSmoker	Paciente es fumador	0: No, 1: Sí
CigPerDay	Número de cigarrillos por día	0-70
BPMeds	Medicación para presión arterial	0: No, 1: Sí
PrevalentStroke	Accidente cardiovascular	0: No, 1: Sí
PrevalentHyp	Paciente hipertenso	0: No, 1: Sí
Diabetes	Si el paciente tiene Diabetes	0: No, 1: Sí
TotChol	Colesterol toral	107-696
SysBP	Presión arterial sistólica	83.5-295
DiaBP	Presión arterial diastólica	48-142.5
BMI	Índice de masa corporal	15.54-56.8
HeartRate	Frecuencia cardíaca	44-143
Glucose	Nivel de glucosa	40-394
TenYearCHD	Riesgo cardiovascular a 10 años	0: No, 1: Sí

SI: Secundaria Incompleta, SC: Secundaria Completa, ET: Educación Técnica, U: Universidad

Tabla 4.5: Descripción de los atributos del conjunto de datos Framingham.

Por otro lado, en la Tabla 4.7 se muestran los estadísticos descriptivos de las características cuantitativas.

Variables	Min	Max	Media	Mediana	Moda	Desv. Est.	Varianza
Age	32	70	49.58	49	40	8.57	7.49
CigPerDay	0	70	9.01	0	0	11.92	142.14
TotChol	107	696	236.70	234	240	44.59	1988.38
SysBP	83.50	295	132.35	128	120	22.03	485.46
DiaBP	48	142.5	82.89	82	80	11.91	141.85
BMI	15.54	56.80	25.80	25.40	22.19	4.07	16.64
HeartRate	44	143	75.88	75	75	12	144.60
Glucose	40	394	81.96	78	75	23.95	573.81

Tabla 4.7: Estadísticos descriptivos de las variables cuantitativas del conjunto de datos Framingham.

Cabe mencionar que el conjunto de datos presenta desbalance de registros con respecto a la variable objetivo (*TenYearCHD*). En la Figura 4.7 vemos que la clase 0 tiene 3596 registros de personas sanas y solo a 644 de la clase 1 se le diagnosticó riesgo cardiovascular.

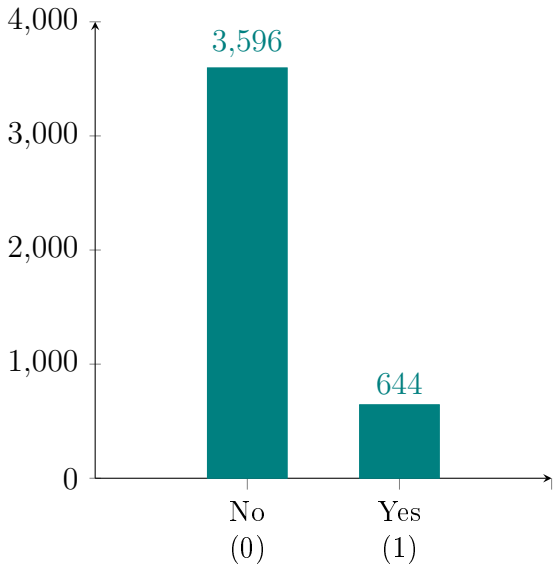


Figura 4.7: Proporción de registros de Framingham en relación con la variable *TenYearCHD*

4.1.2.1 Análisis de correlación

Los atributos que presentan correlación positiva son: presión sistólica (sysBP) - presión diastólica (diaBP), sysBP – edad (age), índice de masa corporal (BMI) – diaBP. Por ejemplo, la primera relación es fuerte, en este caso (Patel et al., 2016) menciona que la presión arterial

es importante para evaluar la salud cardiovascular y puede proporcionar información sobre posibles enfermedades. La Figura 4.8 muestra la matriz con los valores de correlación de las variables cuantitativas.

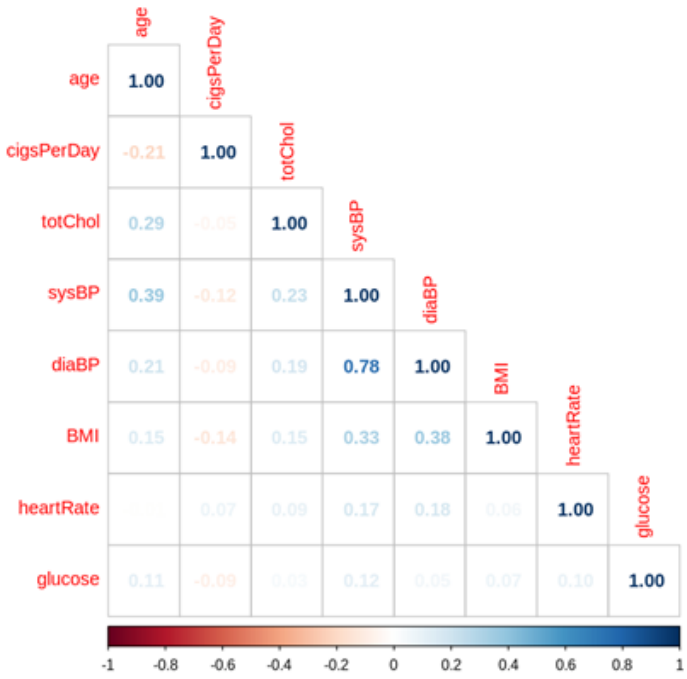
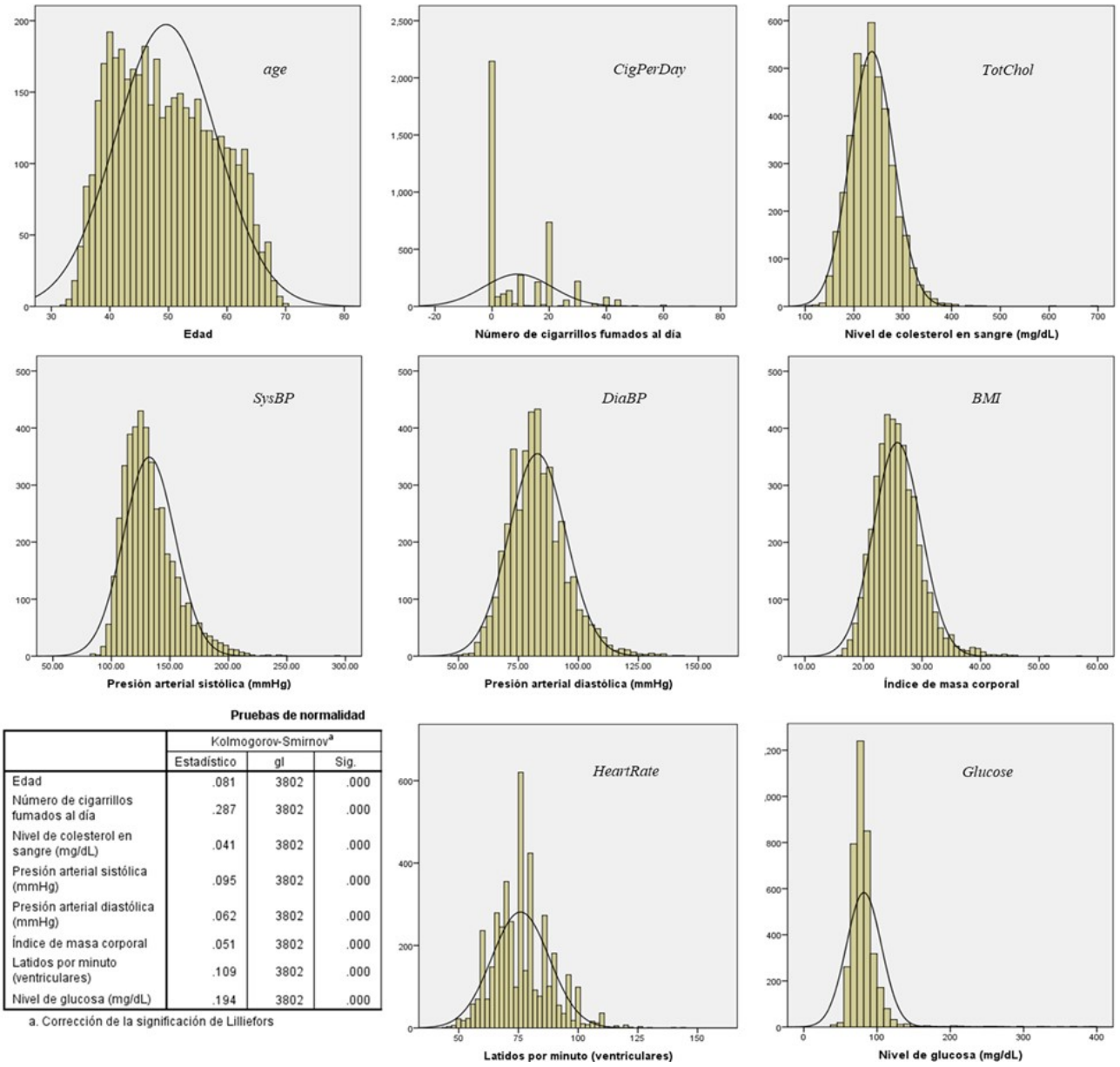


Figura 4.8: Matriz de correlación de las variables cuantitativas del conjunto de datos Framingham.

4.1.2.2 Prueba de normalidad

En la Figura 4.9 se muestran los histogramas de frecuencias de las variables cuantitativas. Las características *cigsPerDay, heartRate* y *glucose* presentan un ligero sesgo a la izquierda. En la Tabla de resultados de la prueba de KS observamos que para el caso de las ocho características el valor de la columna Sig. es menor que 0.05, por lo que la hipótesis nula se rechaza, es decir, no se cuenta con normalidad en la distribución de las ocho características.



Pruebas de normalidad

	Kolmogorov-Smirnov[a]		
	Estadístico	gl	Sig.
Edad	.081	3802	.000
Número de cigarrillos fumados al día	.287	3802	.000
Nivel de colesterol en sangre (mg/dL)	.041	3802	.000
Presión arterial sistólica (mmHg)	.095	3802	.000
Presión arterial diastólica (mmHg)	.062	3802	.000
Índice de masa corporal	.051	3802	.000
Latidos por minuto (ventriculares)	.109	3802	.000
Nivel de glucosa (mg/dL)	.194	3802	.000

a. Corrección de la significación de Lilliefors

Figura 4.9: Histogramas y prueba KS de las variables cuantitativas del conjunto de datos Framingham.

4.1.2.3 Identificación de datos faltantes

El conjunto de datos Framingham presenta datos faltantes en seis de sus características (aproximadamente el 15 % son datos perdidos). En la Figura 4.10 podemos ver una matriz donde se presentan los patrones de datos perdidos y su porcentaje por cada variable. Es importante poner atención en aquellos patrones que relacionan a dos o más variables. Esto nos puede

dar idea del tipo de patrón de dato perdido que se presentan. Por ejemplo: datos perdidos aleatoriamente, datos perdidos no aleatoriamente, o datos perdidos completamente al azar.

En nuestro caso identificamos que el patrón de datos perdidos es de tipo perdidos completamente al azar ya que la ausencia de los datos no está relacionada con ninguna variable y es completamente al azar. Las conclusiones del análisis de los datos perdidos es el siguiente:

- El patrón entre colesterol y glucosa está presente en el 80 % de registros (con datos perdidos) etiquetados con no riesgo cardiovascular.
- El patrón entre cigarrillos por día y glucosa está presente en el 100 % de registros (que presentan datos perdidos) etiquetados con no riesgo cardiovascular. Sin embargo, estas personas son fumadores
- El patrón entre BMI y glucosa está presente en el 50 % de registros (que presentan datos perdidos) etiquetados con riesgo cardiovascular. Estas personas son mujeres no fumadoras y no han tenido un accidente cerebrovascular
- El patrón entre BMI, colesterol y glucosa es solo un registro etiquetado con no riesgo cardiovascular. Esta persona no es fumadora, no toma medicamento para la presión, no ha tenido accidente cerebrovascular, no es hipertensa y no es diabética.

4.1.2.4 Identificación de datos atípicos

La Figura 4.11 muestra un gráfico de cajas y bigotes de las variables cuantitativas, se observa que la mayoría de las características contienen datos atípicos y solo el atributo *age* está libre de ellos. Además, podemos identificar que existe diferencias de medias entre las características, esto ocurre porque las variables tienen escalas distintas.

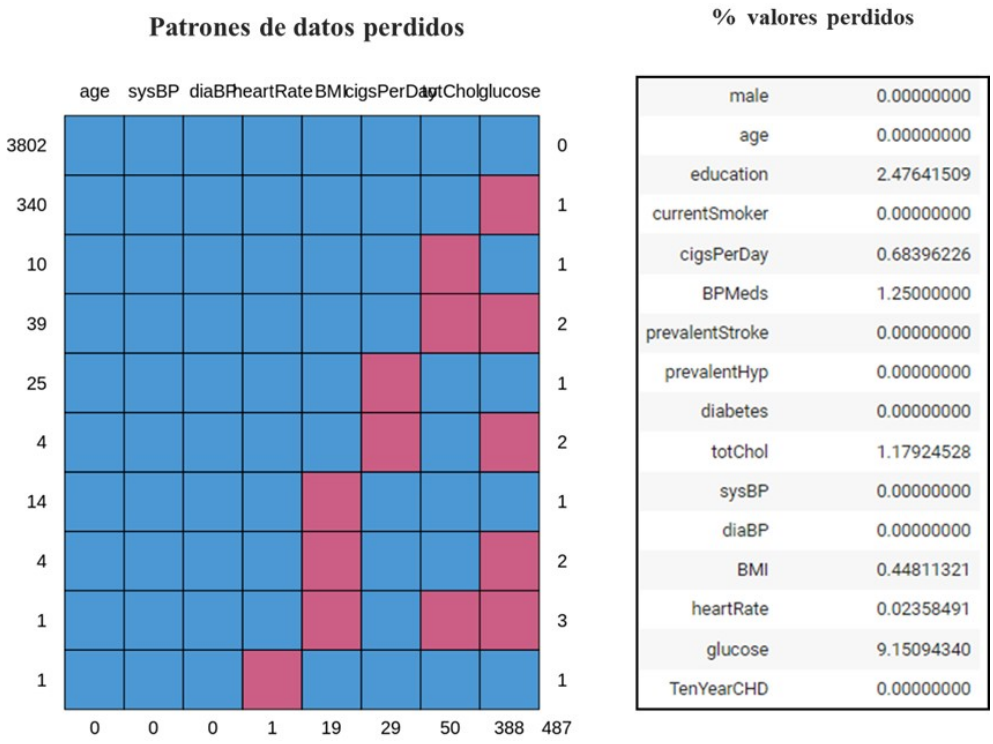


Figura 4.10: Patrones de datos perdidos de las variables cuantitativas del conjunto de datos Framingham.

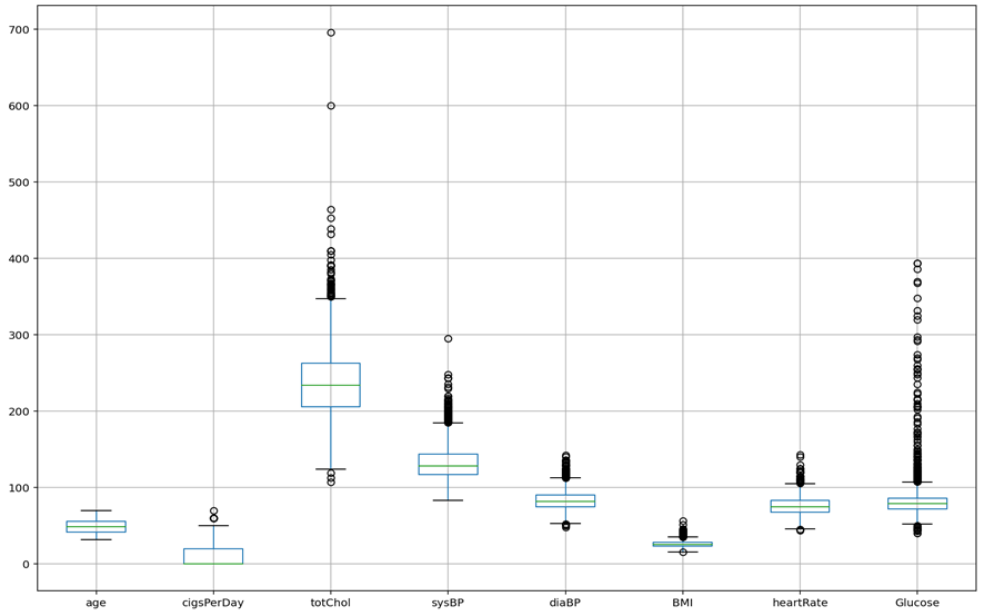


Figura 4.11: Cajas y bigotes de las variables cuantitativas del conjunto de datos Framingham.

4.1.2.5 Preprocesamiento de datos

El mecanismo de tratamiento de los datos perdidos fue a partir de la imputación del estadístico media para el subconjunto de variables cuantitativas, y para las variables cualitativas usamos el estadístico moda. Por otro lado, para el tratamiento de los datos atípicos aplicamos la técnica de Winsorización. Es una técnica que se utiliza para tratar los valores atípicos reemplazándolos por los valores más cercanos dentro de un rango definido. Por ejemplo, los valores que caen en el percentil 1 o 99 pueden ser reemplazados por el valor en el percentil 5 o 95, respectivamente. En nuestro caso, consideramos el 1 % del extremo inferior y el 3 % del extremo superior como datos atípicos. La Figura 4.12 muestra el diagrama de cajas y bigotes sin datos atípicos.

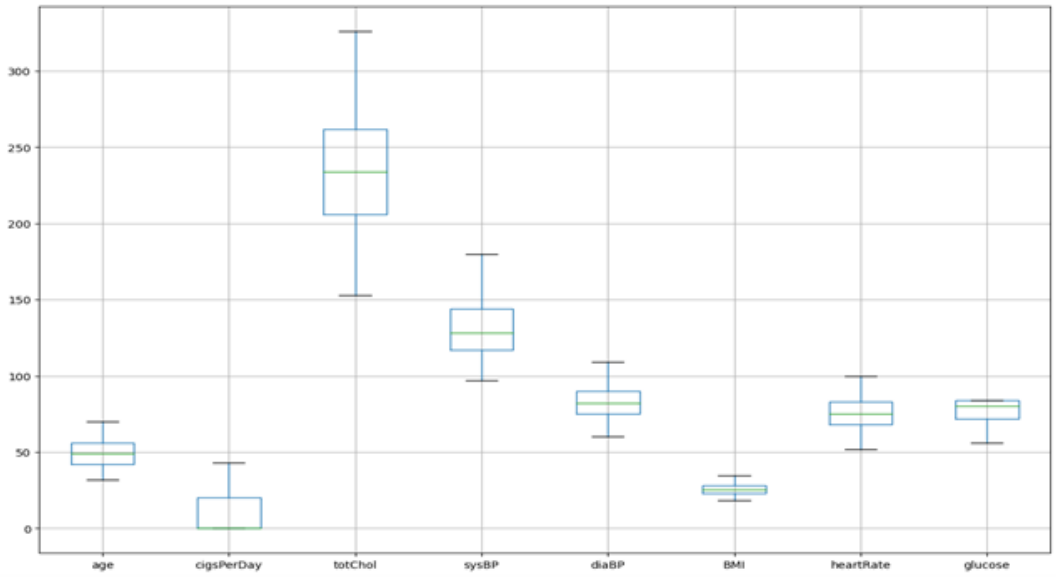


Figura 4.12: Variables cuantitativas del conjunto de datos Framingham sin datos atípicos.

Por otro lado, en el conjunto de datos Framingham también fue necesario una escala unitaria ya que las variables que lo conforman tienen distintas escalas. En el caso de las variables cuantitativas se empleó la técnica *Robust Scaler* para obtener la nueva escala. Además, el conjunto de datos contiene variables cualitativas del tipo ordinaria (education), para su tratamiento se empleó el método de codificación *OneHot*. En la Figura 4.13 se muestra el resultado de la transformación.

education
Secundaria Incompleta
Secundaria Completa
Educación Técnica
Universidad

Secundaria Incompleta	Secundaria Completa	Educación Técnica	Universidad
0	1	0	1
0	0	0	0
1	0	0	0
0	0	1	0

Figura 4.13: Codificación *OneHot* del atributo *education*.

El análisis exploratorio de los datos nos ayudó a identificar un desbalance en las observaciones con respecto a la variable objetivo. Cuando existen más observaciones de una clase que de otra en el conjunto de datos, puede haber varios problemas al desarrollar modelos predictivos. Uno de los problemas al entrenar los algoritmos es que puede ajustarse demasiado a la clase más mayoritaria y no prestar suficiente atención a la clase minoritaria. Además, se pueden emplear métricas de desempeño incorrectas. La métrica *accuracy* no es una medida adecuada para evaluar la calidad del modelo. Se deben utilizar otras métricas, como F1-score para evaluar el modelo de manera más completa (Benhar et al., 2020).

Para solucionar el desbalance de clase aplicamos la técnica de sobremuestreo *Adaptive Synthetic Sampling Approach for Imbalanced Learning* (ADASYN). Es un algoritmo que crea datos artificiales basados en las similitudes del espacio de características de la clase minoritaria. Lo hace seleccionando una instancia aleatoria y luego identifica sus k-vecinos más cercanos. Se generan instancias sintéticas interpolando entre la instancia seleccionada y sus vecinos (ver Figura 4.16).

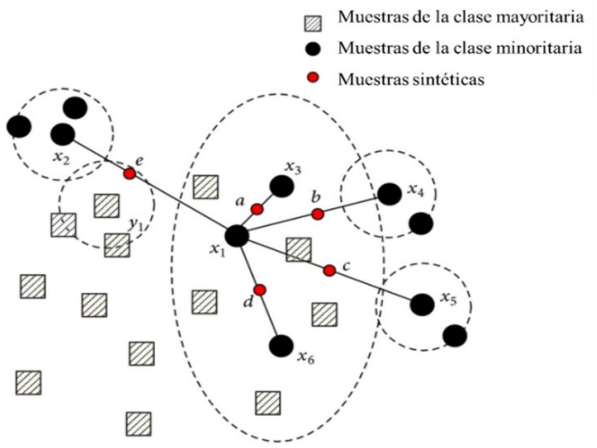


Figura 4.14: Mecanismo de generación de muestras sintéticas mediante la técnica ADASYN .

En la Grafica 4.15 se muestra que la distribución antes de realizar el remuestreo era 3596 observaciones para la clase 0 y 644 de la clase 1. Posteriormente con la técnica de remuestreo ADASYN ambas clases quedaron con 3596 observaciones (ver Gráfico 4.16).

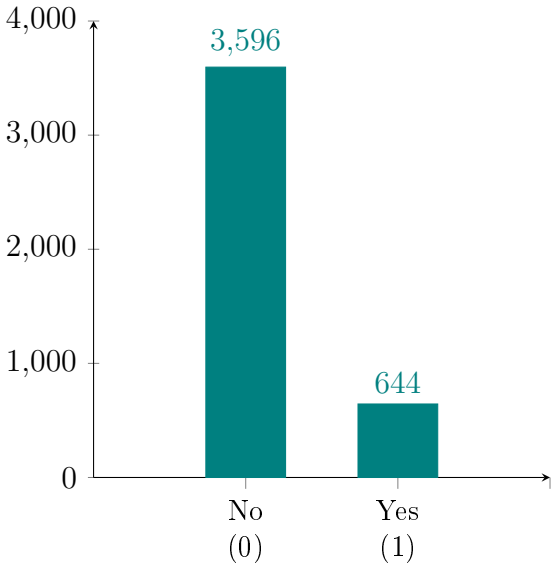


Figura 4.15: Ántes del remuestreo.

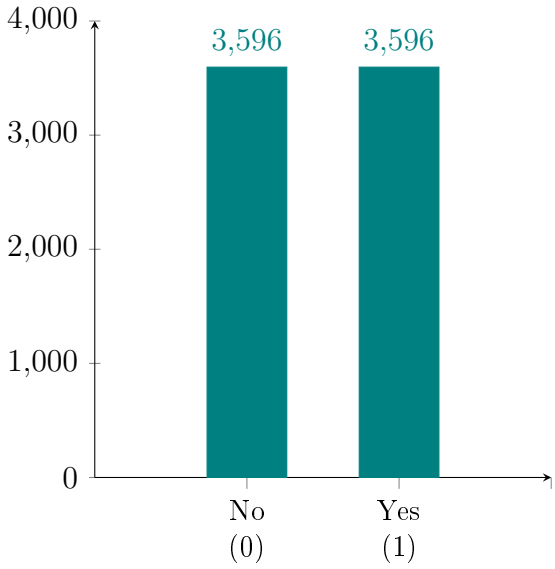


Figura 4.16: con ADASYN.

Durante el análisis exploratorio se ha identificado que los datos médicos son en su mayoría discretos, por lo que estandarizarlos es esencial para hacer converger las características de estos. En el preprocesamiento de este conjunto de datos aplicamos el mecanismo llamado estandarización robusta, utilizando la biblioteca *Scikit-learn*. Es uno de los métodos más populares para definir una única escala (para todas las características) utilizando los percentiles 25 y 75 de la distribución de datos para cada característica.

4.1.3 Conjunto de datos Faisalabad

La Tabla 4.8 describe las 12 características y la variable objetivo del conjunto de datos Faisalabad.

Variables	Descripción	Rango
age	Edad del paciente	40-95
anaemia	Disminución de glóbulos rojos o hemoglobina	0: No, 1: Sí
High_blood_pressure	Si el paciente tiene hipertensión	0: No, 1: Sí
creatinine_phosphokinase	Nivel de la enzima CPK en la sangre	23-7861
diabetes	Si el paciente tiene diabetes	0: No, 1: Sí
ejection_fraction	Porcentaje de sangre que sale del corazón en cada contracción	14-80
sex	sexo de paciente	0: masculino, 1: femenino
platelets	Plaquetas en la sangre	25100-850000
serum_creatinine	Nivel de creatinina en la sangre	0.50-9.40
serum_sodium	Nivel de sodio en la sangre	113-148
smoking	Si el paciente fuma	0: No, 1: Sí
time	Periodo de seguimiento	4-285
death_event	Si el paciente ha fallecido durante el periodo de seguimiento	0: No, 1: Sí

Tabla 4.8: Descripción de los atributos del conjunto de datos Faisalabad.

Existe una proporción desbalanceada de las muestras en relación con el sexo de las personas es del 57 % para mujeres y 43 % de hombres. En la Tabla 4.9 se muestran los estadísticos descriptivos de las características cuantitativas. Podemos observar que este estudio se realizó en personas adultas (40-95 años) y la edad media es de 60 años.

Variables	Min	Max	Media	Mediana	Moda	Desv. Est.	Varianza
age	40	95	60.83	60	60	11.89	141.48
creatinine	23	7861	581.83	250	582	970.28	941458.6
ejection_fraction	14	80	38.08	38	35	11.83	140.06
platelets	25100	850000	263358.02	262000	263358.03	97804.26	9565668749.449
serum_creatinine	0.50	9.40	1.39 1	1.1	1	1.03	1.070
serum_sodium	113	148	136.62	137	136	4.41	19.47
time	4	285	130.26	115	187-250	77.61	6023.96

Tabla 4.9: Estadísticos descriptivos de las variables cuantitativas del conjunto de datoa Faisalabad.

Cabe mencionar que el conjunto de datos presenta desbalance de registros con respecto a la variable objetivo (death_event). La clase 0 tiene 203 registros de personas con insuficiencia cardíaca sin evento de muerte, y 96 registros de personas (clase 1) fallecieron a causa de

insuficiencia cardíaca durante el estudio.

4.1.3.1 Análisis de correlación

La Figura 4.17 presenta el resultado del análisis de la correlación de Spearman, se visualizan los valores que expresan la relación entre distintos pares de variables. Los tonos más intensos en el gráfico denotan una correlación más fuerte entre las variables.

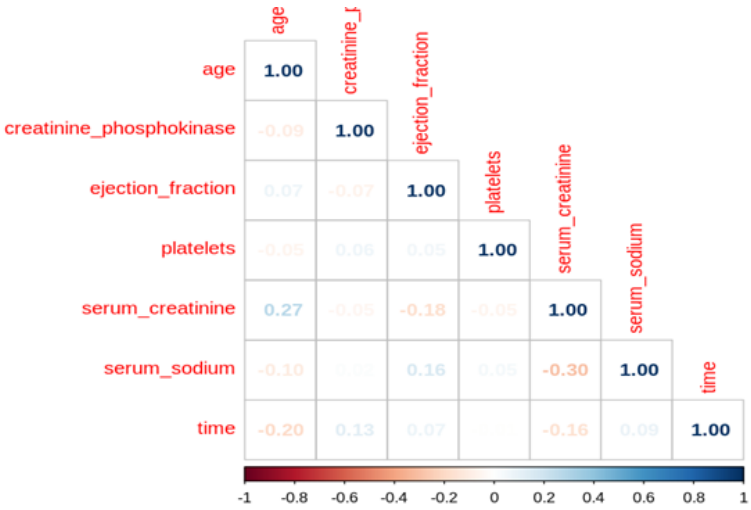
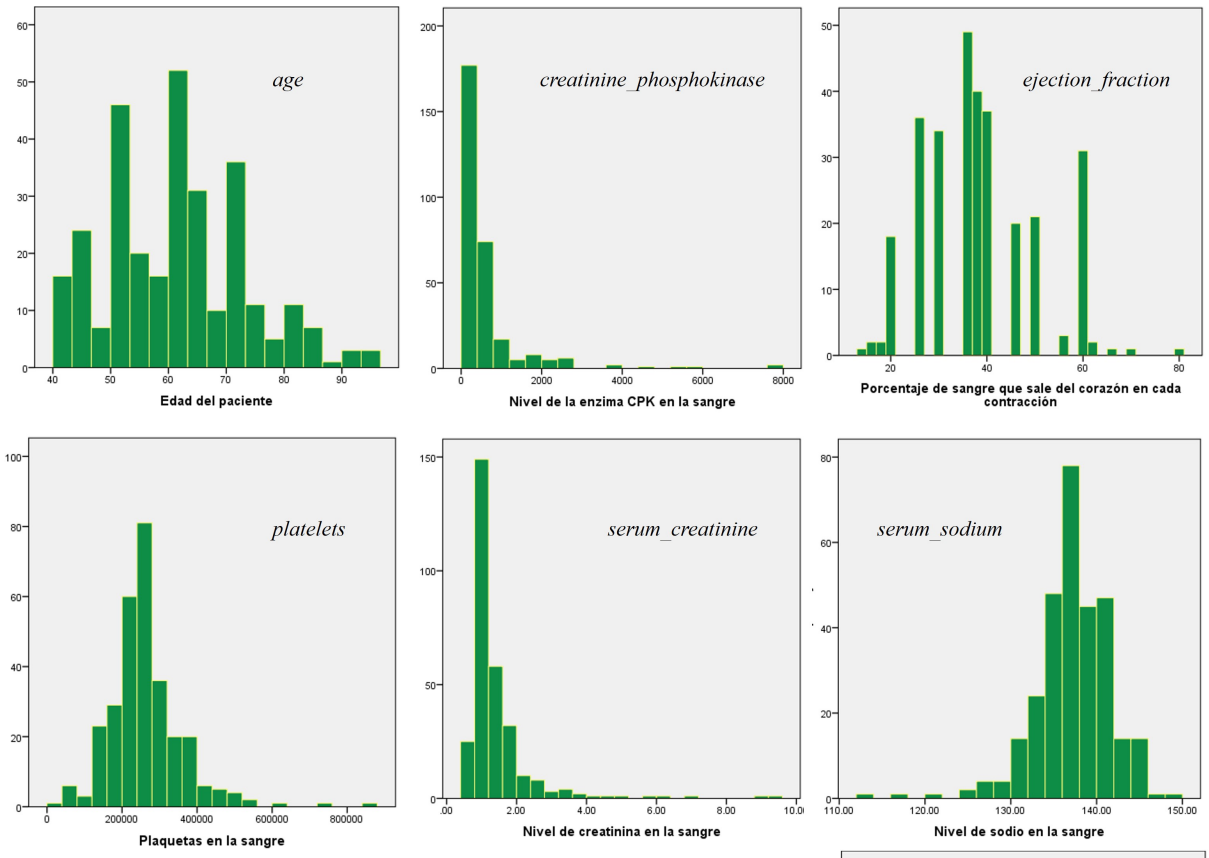


Figura 4.17: Matriz de correlación de las variables cuantitativas del conjunto de datos Faisalabad.

4.1.3.2 Prueba de normalidad

Todas las características presentan un ligero sesgo en diferentes sentidos. La Figura 4.18 muestra los histogramas de frecuencia de las variables cuantitativas del conjunto de datos Faisalabad. Además, en la Tabla de resultados de la prueba de KS observamos que para el caso de las ocho características el valor de la columna Sig. es menor que 0.05, por lo que la hipótesis nula se rechaza, es decir, no se cuenta con normalidad en la distribución de las ocho características.



Tests of Normality

	Kolmogorov-Smirnov[a]		
	Statistic	df	Sig.
Edad del paciente	.070	299	.001
Nivel de la enzima CPK en la sangre	.287	299	.000
Porcentaje de sangre que sale del corazón en cada contracción	.168	299	.000
Plaquetas en la sangre	.116	299	.000
Nivel de creatinina en la sangre	.265	299	.000
Nivel de sodio en la sangre	.113	299	.000
Periodo de seguimiento	.105	299	.000

a. Lilliefors Significance Correction

Figura 4.18: Histogramas de frecuencia y prueba KS de las variables cuantitativas del conjunto de datos Faisalabad.

4.1.3.3 Identificación de datos faltantes

En el conjunto de datos Faisalabad no se identificaron datos atípicos, así lo mostró el resultado de la consulta aplicada. En la Figura 4.19 se ilustra el resultado.

```
In [12]: # Ver la cantidad de valores nulos en cada una de las características
         datosFaisalabad.isna().sum()

Out[12]: age                       0
         anaemia                   0
         creatinine_phosphokinase  0
         diabetes                  0
         ejection_fraction         0
         high_blood_pressure       0
         platelets                 0
         serum_creatinine          0
         serum_sodium              0
         sex                       0
         smoking                   0
         time                      0
         DEATH_EVENT               0
         dtype: int64
```

Figura 4.19: Resultado de la consulta de datos atípicos del conjunto de datos Faisalabad.

4.1.3.4 Identificación de datos atípicos

Por medio del gráfico de cajas y bigotes (ver Figura 4.20) se observó la presencia de datos atípicos en la mayoría de las variables. Para el caso del atributo *platelets* el número de puntos atípicos es mayor, esto se debe a que la escala de este atributo es mucho más grande.

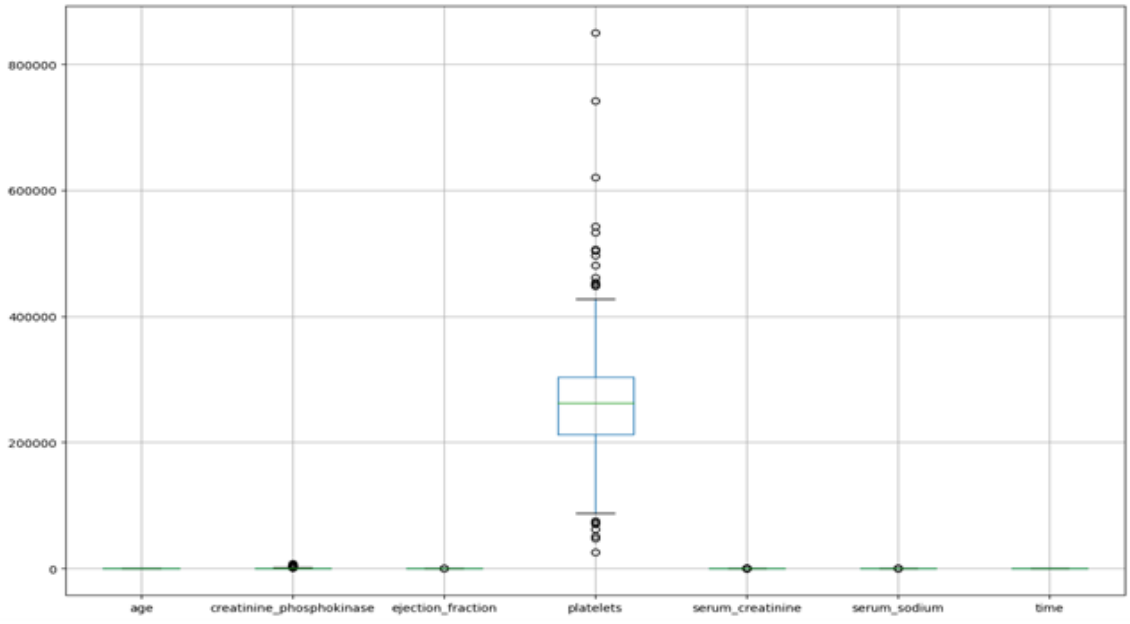


Figura 4.20: Cajas y bigotes de las variables cuantitativas del conjunto de datos Faisalabad con datos atípicos.

4.1.3.5 Preprocesamiento de datos

Se realizó el proceso de balance de clases: al igual que el conjunto de datos Framingham, se aplicó la técnica ADASYN. Logrando que ambas clases quedaran con la misma cantidad de registros (203 registros aprox.)

Se aplicó la técnica de Robust Scalert para crear una escala unitaria, esta decisión es un procedimiento que no se pudo omitir en el preprocesamiento del conjunto de datos Faisalabad. La Figura 4.21 muestra el diagrama de cajas y bigotes, observamos que todos los atributos tienen la misma media, eso indica que están sobre la misma escala. Además, aun podemos observar la presencia de los datos atípicos.

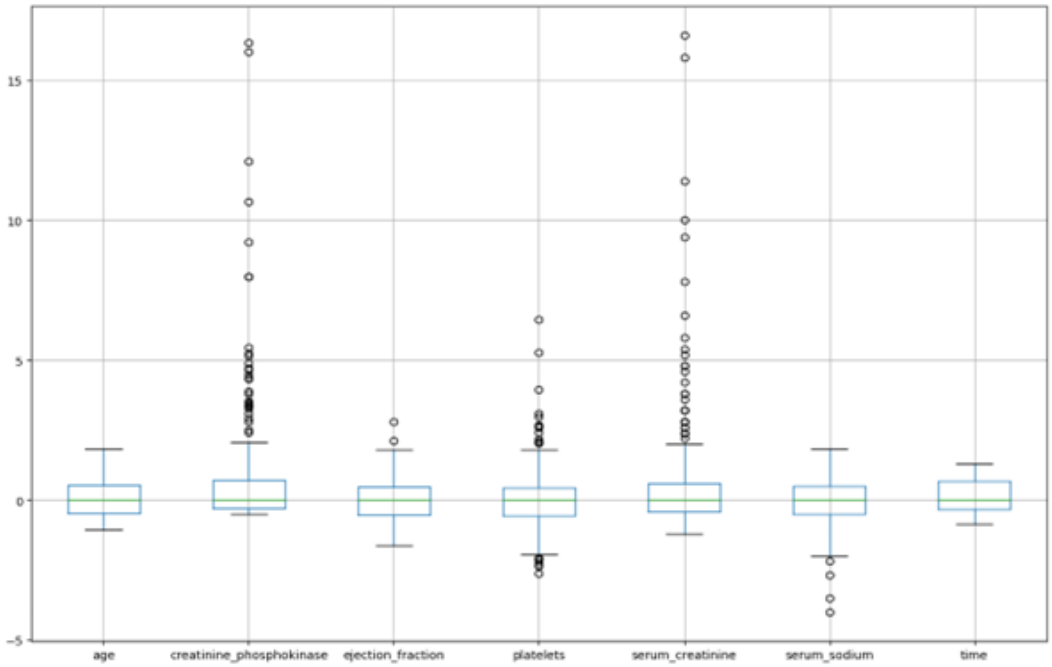


Figura 4.21: Variables cuantitativas del conjunto de datos Faisalabad escalados.

Se realizó la división de los conjuntos de datos tanto para la fase de entrenamiento como para la fase prueba. El total de registros y su porcentaje correspondiente se muestra en la Tabla 4.11.

Conjunto de datos	Entrenamiento	Prueba	Folds - CV
Cleveland	242 (80 %)	61 (20 %)	10
Framingham	6972 (93 %)	220 (7 %)	10
Faisalabad	328 (80 %)	83 (20 %)	10

Tabla 4.11: Distribución de registros destinados para las fases del modelado.

4.4. Definición de los procedimientos de optimización

Para el proceso de entrenamiento/optimización de los modelos se utilizó el lenguaje de programación Python y las bibliotecas *ScikitLearn* y *Optunity* que contienen los algoritmos de aprendizaje automático. En cuanto al hardware, se utilizó un ordenador HP-Victus con procesador Core™ i5-12450H a 2,00 GHz, 24 GB de RAM y sistema operativo Windows 11.

4.4.1. Inicialización de parámetros de las metaheurísticas

La asignación de valores a los parámetros de los enfoques PSO y GA es un reto debido a la complejidad de estos métodos y la naturaleza multidimensional de los espacios de búsqueda (Yang y Shami, 2020). La Tabla 4.13 muestra los valores iniciales sugeridos que se utilizaron para refinar y controlar el proceso de búsqueda de los enfoques de optimización.

Enfoque de optimización	Parámetro	Valor
Particle Swarm Optimization	Tamaño de la población	50
	Constante de inercia w	$[0{,}5 + (rand/2{,}0)]$
	Factores de aprendizaje c_1 y c_2	$c_1 = 2{,}8, c_2 = 1{,}3$
Genetic Algorithm	Tamaño de la población	50
	Tasa de cruce	0.6
	Tasa de mutación	0.10
	Número de generaciones	10
	Tamaño del torneo	3

Tabla 4.13: Inicialización de los parámetros de los enfoques PSO y GA.

4.4.2. Definición del espacio de búsqueda

Se propuso una conjunto de valores del espacio de búsqueda para cada algoritmo de aprendizaje automático, estos valores serán la entrada que procesará cada mecanismo de optimización. Los enfoques de optimización aplicarán sus propios mecanismos de búsqueda y evaluarán las soluciones (modelos), este procedimiento se repetirá hasta obtener la mejor solución teniendo como referencia alguna métrica de desempeño. En esta investigación se propuso la métrica curva ROC-AUC, ya que nos interesa que el modelo pueda distinguir las clases del conjunto de datos.

A continuación, se mencionan algunas consideraciones a la hora de definir el espacio de búsqueda.

- Existe diferencia significativa en la cantidad de registros presentes en los conjuntos de datos. El conjunto de datos Cleveland contiene 303 registros, mientras que Framingham es considerablemente más grande. Debido a esta variabilidad se decidió proponer un rango de valores más pequeño para los hiperparámetros en el caso del conjunto de datos Cleveland, con el fin de evitar el sobreajuste y favorecer una mayor generalización del modelo.

- La variabilidad que existe en las características de los datos, ya que cada conjunto de datos presenta diferentes distribuciones y patrones.

- Los recursos computacionales disponibles también influyen en la propuesta de los espacios de búsqueda, un espacio de soluciones más amplio podría implicar mayor costo de procesamiento.

En las Tablas 4.15, 4.17 y 4.19 se presentan los rangos de valores válidos en el proceso de búsqueda de la mejor combinación de los hiperparámetros en la fase de entrenamiento, considerando los conjuntos de datos de Cleveland, Framingham y Faisalabad, respectivamente.

Clasificador	Hiperparámetro	Grid Search	Random Search	Bayesian Opt.	Particle S. O.	Genetic Alg.
RF	n_estimators	15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 70, 80, 90, 100	20,22,25, 28, 30, 32, 34, 36, 38, 40, 45, 50, 52, 54, 56, 58, 60	20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90	[50 - 100]	[20 - 150]
	max_depth	4, 5, 7, 9	5, 7, 10, 12	5, 7, 9, 10, 12, 15	[5 - 10]	[5 - 15]
	min_samples_leaf	2, 3, 4, 5, 6	3, 4, 5, 6, 7	1, 2, 3, 4, 5, 6, 7	[4 - 11]	[4 - 20]
	min_samples_split	-	2, 3, 4, 5, 6, 7	1, 2, 3, 4, 5	[4 - 11]	[4 - 20]
SVM	C	1,2, 2.5, 6, 7, 10, 11, 11.5, 12.5, 13-18, 20	2,3, 4,5, 6,7, 8,9, 10, 15, 20, 25	2, 2.5, 3,4, 5, 10.5, 11.5, 12.3, 10-18, 20	[1 - 40]	[20- 100]
	gamma	'scale', 1, 0.1, 0.01, 0.001, 0.09, 0.0001, 0.004, 0.05, 0.3	1, 0.1, 0.01, 0.09, 0.05, 0.3, 0.0004, 0.003	1, 0.1, 0.02, 0.001, 0.004, 0.3, 0.012	[0.0001 - 1.0]	[0.0001 - 1]
	kernel	'linear','poly','rbf'	'poly','rbf','sigmoid'	'poly','rbf'	'linear', 'poly', 'rbf', 'sigmoid'	'linear', 'poly', 'rbf', 'sigmoid'
	coef0	-	1, 1.5, 2.4, 3	0.1, 2, 0.5, 0.23, 1	[-1 - 1]	[-1 - 3]
	degree	1, 2, 3	2, 4, 5	1, 2, 3, 4, 5	[1 - 6]	[1 - 6]
XGBoost	n_estimators	15, 25,27,30, 40, 45, 50, 70, 75, 80	5, 10, 15, 20, 25, 30, 35, 40, 60, 65, 70, 90	20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70	[20 - 150]	[20 - 120]
	learning_rate	0.01, 0.02, 0.1, 0.08	0.01, 0.03, 0.06, 0.1, 0.15, 0.2, 0.25, 0.4, 0.5	0.01, 0.03, 0.15, 0.2, 0.25, 0.4, 0.05, 0.6, 0.7	[0.01 - 2]	0.001, 0.01, 0.1, 0.2, 0.3
	gamma	0.5, 1, 1.5, 2, 3	0.1, 0.2, 0.002, 0.4, 0.8, 1.5, 2, 3.5	0.2, 0.003, 0.4, 0.16, 0.8, 1.6, 3.2, 3.4	[0.1 - 2]	[0.1 − 0.4]
	subsample	0.6, 0.8, 1	0.6, 0.8, 0.9, 1	0.6, 0.7, 0.8, 0.9, 1.0	[0.5 - 1]	[0.7 - 1]
	max_depth	3, 4, 5	5, 6, 7, 8	3, 4, 5, 6, 7, 8, 10	[3 - 20]	[5 - 20]
	colsample_bytree	0.5, 0.7, 0.8, 1	0.6, 0.7, 0.8, 0.9	0.6, 0.8, 1	[0.3 - 1]	[0.6 - 1]
	min_child_weight	-	1, 2, 3, 4, 5, 10	4, 5, 10	[1 - 10]	[1 - 10]

Tabla 4.15: Espacio de búsqueda para entrenar con el conjunto de datos Cleveland.

Clasificador	Hiperparámetro	Grid Search	Random Search	Bayesian Opt.	Particle S. O.	Genetic Alg.
RF	n_estimators	80, 90, 120, 130, 150	50, 75, 100, 125, 150, 200, 225, 250	100, 110, 120, 130, 140, 150, 200, 210, 225, 230, 240, 250	[150 - 300]	[80 - 250]
	max_features	0.5, 0.7, 0.9	0.5, 0.7, 0.9, 1	0.5, 0.7, 0.9, 1	0.4 - 1	0.3 -1
	max_depth	3, 5, 7, 10	5, 7, 8, 9, 10, 11, 12, 15, 30	4, 5, 7, 8, 9, 10, 11, 12, 15, 20, 25, 30	[5 - 30]	[5-25]
	min_samples_split	2	4, 5, 6, 10, 12, 15	4, 5, 6, 10, 12, 15, 20	[2 - 15]	[4 - 20]
	min_samples_leaf	1	4, 5, 6, 10, 12, 15	4, 5, 6, 10, 12, 15, 20	[1 - 10]	[4 - 20]
SVM	c	1, 5, 10, 15, 20, 30, 40, 50, 60	2, 2.5, 3, 5, 7, 10, 12, 14, 15, 17, 18, 20	1, 2, 2.5, 3,4, 5, 7, 10, 11, 13, 14, 15	[1 - 60]	[40 - 100]
	kernel	'linear', 'poly', 'rbf'	'poly', 'rbf', 'sigmoid'	'linear', 'poly', 'rbf', 'sigmoid'	'poly', 'rbf', 'sigmoid'	'poly', 'rbf'
	degree	3	[2 - 5]	[1 - 6]	[1 - 5]	[1 - 6]
	gamma	'scale'	1, 0.1, 0.01, 0.001, 0.0001, 0.05, 0.02	1, 1.5, 0.01, 0.2, 0.3, 0.4, 0.05	[0.0001 - 1.0]	[0.0001 - 1.0]
	coef0	0	0	0	[-1 - 1]	[-1 - 3]
XGBoost	n_estimators	80, 110, 150, 170	65, 70, 75, 80, 100, 120, 150, 200	40, 50, 60, 70, 80, 90, 100, 150, 200	[100 - 250]	[20 - 180]
	learning_rate	0.1, 0.3, 0.5, 0.7	0.01, 0.03, 0.06, 0.1, 0.15, 0.2, 0.25, 0.3	0.03, 0.06, 0.1, 0.15, 0.2, 0.25, 0.4, 0.5	[0.01 - 1]	0.001, 0.01, 0.1, 0.2, 0.12, 0.3
	gamma	0	0	0	0	0.1, 0.3, 0.4, 0.23, 0.002
	subsample	0.6, 0.8, 1	0.6, 0.8, 0.9, 1	0.6, 0.8, 1	[0.1 - 1]	0.6, 0.7, 0.8, 0.9, 1
	max_depth	3, 5, 10	3, 4, 5, 6, 7, 8	3, 4, 5, 6, 7, 8	[3 - 25]	[5 - 15]
	colsample_bytree	0.3, 0.6, 0.8	0.5, 0.6, 0.9, 1	0.6, 0.8, 1	[0.5 - 1]	0.6, 0.7, 0.8, 0.9, 1
	min_child_weight	1	1	1	1	[0.1 - 2]

Tabla 4.17: Espacio de búsqueda para entrenar con el conjunto de datos Framingham.

Clasificador	Hiperparámetro	Grid Search	Random Search	Bayesian Opt.	Particle S. O.	Genetic Alg.
RF	n_estimators	50, 80, 130, 135, 140, 145, 150	20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90	20, 40, 50, 75, 90, 120, 140	[20 - 120]	[20 - 120]
	max_features	0.5, 0.7, 0.9, 'auto'	0.8, 0.9, 1	0.7, 0.9, 1	[0.7 - 1]	-
	max_depth	2, 3, 5, 7, 10, 12, 15	5, 7, 8, 9, 10	2, 4, 5, 7, 8, 9, 'auto'	[5,15]	[5,15]
	min_samples_leaf	-	2, 3, 5,7	2, 4, 5, 6, 10	[2 - 8]	[4 - 10]
	min_samples_split	-	2, 4, 6 2, 5, 6, 10	[2 - 12]	[4 - 10]	
SVM	C	1, 3, 4, 4. 5, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25	2, 3.5, 4, 5, 6, 7, 8, 9, 10,11, 12, 13, 14, 15, 16, 17,18, 19, 20	1, 2, 2.5, 3, 4, 5, 5.5, 9.5, 10, 10.5, 11	[1 - 20]	[20 - 100]
	gamma	-	-	1, 0.1, 0.01, 0.001, 0.004, 0.002	[0.0001 - 1]	[0.0001 - 1]
	kernel	'linear', 'poly'	'poly', 'rbf', 'sigmoid'	'linear', 'poly', 'rbf', 'sigmoid'	'linear', 'poly', 'rbf', 'sigmoid'	'linear', 'poly', 'rbf', 'sigmoid'
	Coef0	-	-	-	-	[-1 - 3]
	degree	1, 2, 3 1, 2, 3, 4, 5	1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6	
XGBoost	n_estimators	20, 30, 40, 45, 50, 60, 70	40, 45, 50, 55, 60, 65, 70, 75, 80, 100	20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 150	[20 - 120]	[20 - 120]
	learning_rate	0.1, 0.12, 0.2, 0.3, 0.03	0.1, 0.15, 0.2, 0.25, 0.4, 0.5, 0.6, 0.7	0.01, 0.1, 0.15, 0.2, 0.25, 0.4, 0.5, 0.6	[0.01 - 2]	0.001, 0.01, 0.1, 0.2, 0.12, 0.3
	subsample	0.5, 0.7, 0.9	0.6, 0.8, 1.0	0.5, 0.6, 0.7, 0.8, 1	[0.7 - 1]	0.6, 0.7, 0.8, 0.9, 1
	min_child_weight	-	-	-	-	[1 - 10]
	max_depth	3, 5, 7, 10	4, 5, 6, 7, 8	5, 6, 7, 8, 13, 15, 20	[3 - 10]	[5 - 15]
	gamma	-	-	-	-	0.1, 0.3, 0.4, 0.02
	colsample_bytree	0.7, 0.8, 0.9, 1	0.6, 0.8, 1.0	0.5, 0.6, 0.7, 0.8, 1	[0.01 - 2]	0.6, 0.7, 0.8, 0.9, 1

Tabla 4.19: Espacio de búsqueda para entrenar con el conjunto de datos Faisalabad.

Capítulo 5

Resultados y discusiones

Este capítulo muestra los resultados del proceso de optimización de hiperparámetros de los modelos *Random Forest, Support Vector Machine y Extreme Gradient Boosting* son presentados en esta sección. El proceso de optimización de los modelos se realizó por cada conjunto de datos. Además, se evaluaron los modelos utilizando las métricas de desempeño presentadas anteriormente, tanto en la fase de entrenamiento como en la fase de prueba de los modelos. Además, se analizaron con detalle la matriz de confusión de cada modelo evaluado. Posteriormente, se determinó cuantitativamente el desempeño de cada modelo utilizando las métricas de *accuracy, precision, recall, specificty, f1-score*. Finalmente, se analizó el desempeño de los modelado entrenados y evaluados los tres conjuntos de datos, respectivamente. El objetivo consistió en contrastar las configuraciones de hiperparámetros resultantes y determinar el enfoque de optimización que proporciona mejores resultados. Además, se proporcionó un análisis de los valores que toman los hiperparámetros por cada conjunto de datos

5.1. Evaluación de los modelos RF, SVM y XGBoost

5.1.1 Optimización de hiperparámetros con base en Cleveland

La Tabla 5.2 muestra la matriz de confusión de los modelos de aprendizaje automático con mayor predicción de instancias clasificadas correctamente. Podemos observar que el modelo RF determinó la mayor cantidad de instancias etiquetadas con enfermedad cardiovascular, gracias a la configuración de los hiperparámetros proporcionada por el enfoque RS (renglón sombreado).

El interés de nuestro trabajo radica en la importancia de poder crear modelos con la capacidad de poder identificar a personas con enfermedad cardiovascular. Por ello, en la Figura 5.1 se observa una gráfica del desempeño de los modelos con base en la métrica *Recall*. Los modelos RF y XGBoost presentaron mayor capacidad para identificar los casos con enfermedad cardiovascular (con la mayoría de los enfoques de optimización) y, por lo tanto, superan en desempeño al modelo construido con la configuración predeterminada de los hiperparámetros.

Por otro lado, la visualización de los resultados mediante gráficos de barra son un aspecto esencial en la evaluación de modelos predictivos, ya que desempeñan un papel crucial al proporcionar una representación visual intuitiva de cómo se desempeñan los modelos. Su simplicidad y accesibilidad permiten que personas de diferentes niveles de experiencia, incluso aquellos sin conocimientos profundos en estadísticas o aprendizaje automático, puedan comprender fácilmente la efectividad de un modelo. En ese sentido, la Tabla 5.4 presenta los resultados de la evaluación del desempeño de los modelos generados, utilizando las métricas que se describieron anteriormente (sección 2.2). Los renglones sombreados corresponden a los modelos predictivos que obtuvieron la mayor capacidad de predicción, principalmente en las instancias etiquetados con enfermedad cardiovascular (métrica *recall*).

Algoritmo	Enfoque	Verdaderos Negativos	Falsos Positivos	Falsos Negativos	Verdaderos Positivos	ICC*
	Defecto	28	4	5	24	52
	GS	29	3	2	27	56
RF	**RS**	**30**	**2**	**2**	**27**	**57**
	BO	29	3	2	27	56
	PSO	29	3	3	26	55
	GA	29	3	3	26	55
	Defecto	28	4	4	25	53
	GS	29	3	3	26	55
SVM	RS	29	3	3	26	55
	BO	29	3	3	26	55
	PSO	29	3	3	26	55
	GA	29	3	3	26	55
	Defecto	26	6	3	26	52
	GS	29	3	2	27	56
XGBoost	RS	29	3	2	27	56
	BO	29	3	2	27	56
	PSO	29	3	2	27	56
	GA	30	2	3	26	56

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm, ICC: Instancias Clasificadas Correctamente.

Tabla 5.2: Valores de la matriz de confusión de los modelos evaluados con el subconjunto de datos de Cleveland.

Los gráficos nos ayudan en la comparación de varios modelos; permiten visualizar de manera directa la tendencia del desempeño de cada modelo. Por ello, la Figura 5.2 muestra un gráfico de barras del comportamiento de los modelos, nos ayuda a representar los resultados y facilita la identificación del modelo con mejor desempeño. En términos generales se observa una tendencia en el aumento del desempeño de los modelos RF, SVM y XGBoost.

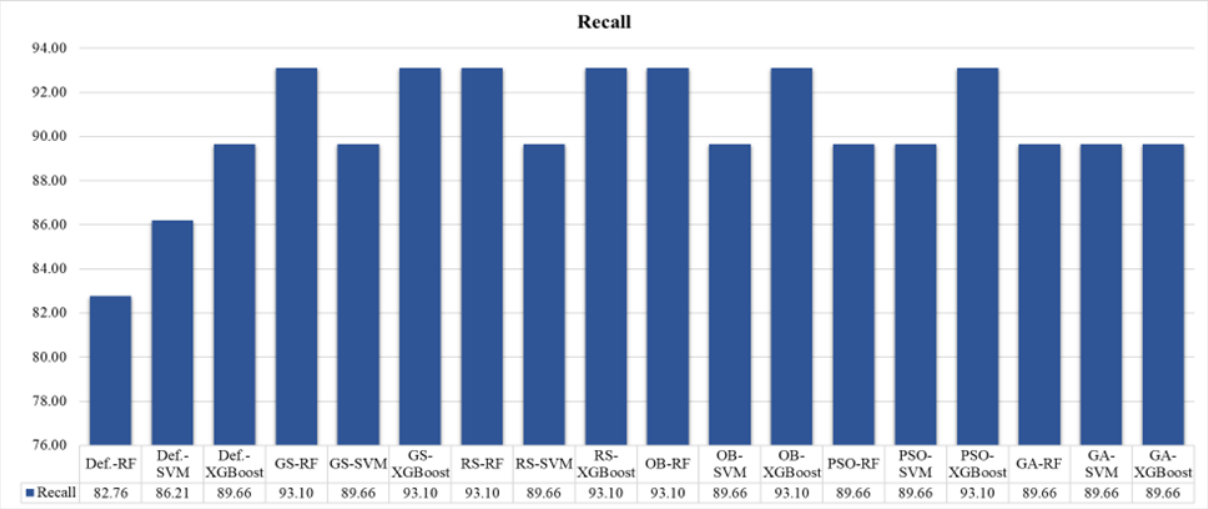


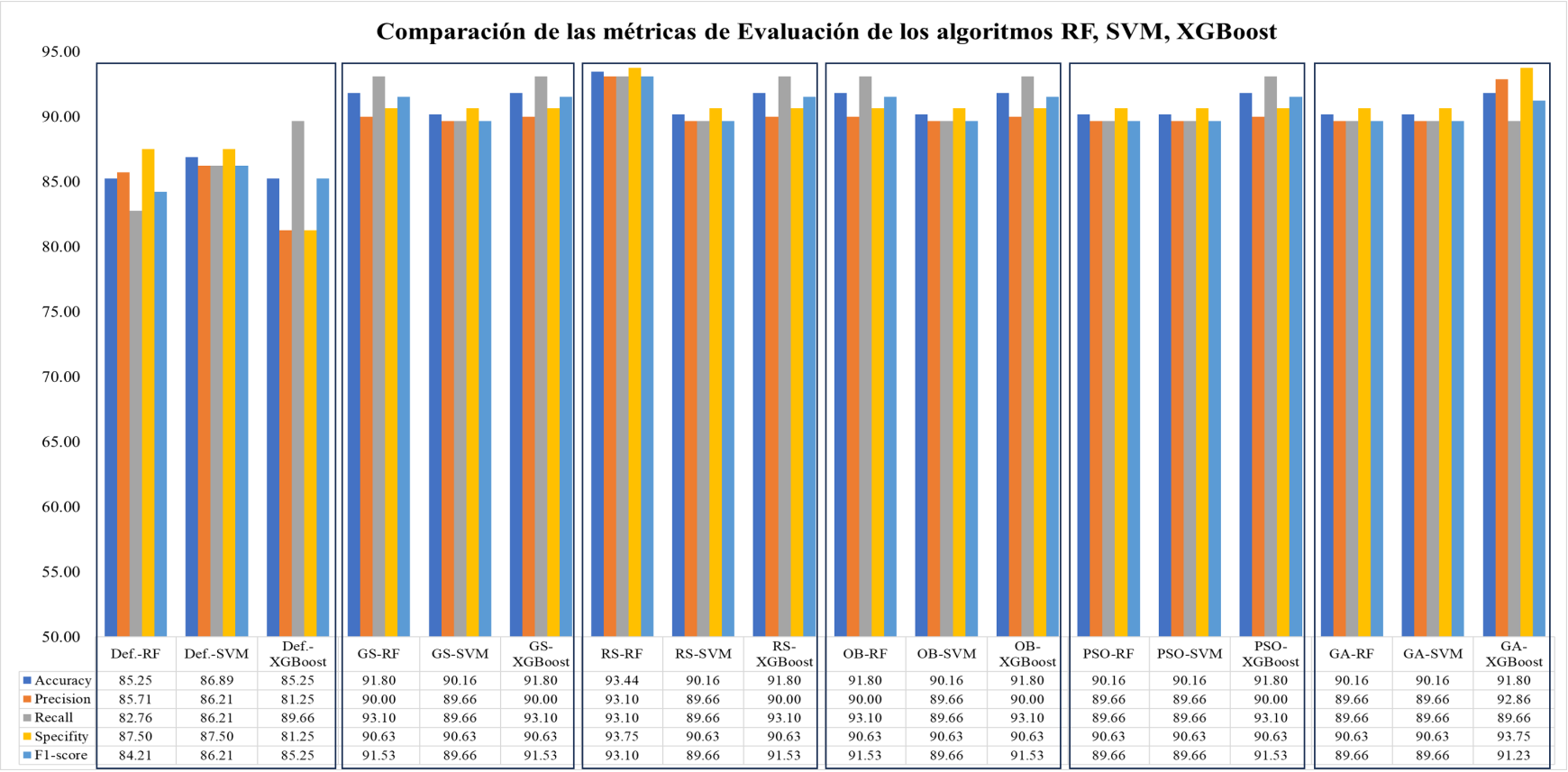
Figura 5.1: Desempeño de los modelos en relación con la métrica *Recall*, evaluados con Cleveland.

Algoritmo	Enfoque	Accuracy train	Accuracy test	Precision	Recall	Specifity	F1-Score
	Defecto	79.35	85.25	85.71	82.76	87.50	84.21
	GS	84.91	91.80	90.00	93.10	90.63	91.53
RF	RS	87.99	93.44	93.10	93.10	93.75	93.10
	BO	82.16	91.80	90.00	93.10	90.63	91.53
	PSO	80.96	90.16	89.66	89.66	90.63	89.66
	GA	80.18	90.16	89.66	89.66	90.63	89.66
	Defecto	81.85	86.89	86.21	86.21	87.50	86.21
SVM	GS	83.90	90.16	89.66	89.66	90.63	89.66
	RS	83.06	90.16	89.66	89.66	90.63	89.66
	BO	83.09	90.16	89.66	89.66	90.63	89.66
	PSO	80.55	90.16	89.66	89.66	90.63	89.66
	GA	83.86	90.16	89.66	89.66	90.63	89.66
	Defecto	76.81	85.25	81.25	89.66	81.25	85.25
	GS	84.43	91.80	90.00	93.10	90.63	91.53
XGBoost	RS	85.78	91.80	90.00	93.10	90.63	91.53
	BO	84.02	91.80	90.00	93.10	90.63	91.53
	PSO	83.88	91.80	90.00	93.10	90.63	91.53
	GA	82.65	91.80	92.86	89.66	93.75	91.23

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.4: Desempeño de los modelos evaluados con el subconjunto de datos de Cleveland.

Los modelos RF y XGBoost presentan mayor capacidad para identificar los casos que presentan enfermedad cardiovascular y por lo tanto superan en desempeño al modelo construido con la configuración predeterminada de los hiperparámetros. Es importante mencionar que la mayoría de los enfoques de optimización mostraron buenos resultados (GS, RS, BO y PSO).



	Def.-RF	Def.-SVM	Def.-XGBoost	GS-RF	GS-SVM	GS-XGBoost	RS-RF	RS-SVM	RS-XGBoost	OB-RF	OB-SVM	OB-XGBoost	PSO-RF	PSO-SVM	PSO-XGBoost	GA-RF	GA-SVM	GA-XGBoost
Accuracy	85.25	86.89	85.25	91.80	90.16	91.80	93.44	90.16	91.80	91.80	90.16	91.80	90.16	90.16	91.80	90.16	90.16	91.80
Precision	85.71	86.21	81.25	90.00	89.66	90.00	93.10	89.66	90.00	90.00	89.66	90.00	89.66	89.66	90.00	89.66	89.66	92.86
Recall	82.76	86.21	89.66	93.10	89.66	93.10	93.10	89.66	93.10	93.10	89.66	93.10	89.66	89.66	93.10	89.66	89.66	89.66
Specifity	87.50	87.50	81.25	90.63	90.63	90.63	93.75	90.63	90.63	90.63	90.63	90.63	90.63	90.63	90.63	90.63	90.63	93.75
F1-score	84.21	86.21	85.25	91.53	89.66	91.53	93.10	89.66	91.53	91.53	89.66	91.53	89.66	89.66	91.53	89.66	89.66	91.23

Figura 5.2: Resultado de las métricas de desempeño de los modelos evaluados con el subconjunto de datos de Cleveland.

En relación con los gráficos de las curvas ROC-AUC de los modelos RF, SVM y XGBoost, en la Figura 5.3 podemos observar su capacidad para distinguir entre pacientes con enfermedad cardiovascular y aquellos que no la presentan. Las curvas ROC muestran una tendencia positiva, lo que indica que los modelos tienen un rendimiento mucho mejor que el azar en la mayoría de los umbrales. El área bajo la curva de los modelos RF, SVM y XGBoost sobrepasa el 0.90, lo que demuestra que los modelos tienen una buena capacidad de discriminación. Cada gráfico representa un enfoque de optimización aplicado a los tres modelos de aprendizaje automático.

La configuración de hiperparámetros de los modelos RF, SVM y XGBoost entrenados con el subconjunto de datos Cleveland se muestran en la Tabla 5.5. Estas configuraciones permiten a los modelos alcanzar su máximo desempeño, en consideración con el espacio de búsqueda propuesto. Se observa en la tabla que los valores de algunos hiperparámetros varían según la técnica de optimización aplicada. Sin embargo, otros hiperparámetros mantienen sus valores predeterminados.

Los resultados del proceso de optimización de los cinco enfoques de optimización difieren de la configuración predeterminada de cada modelo. Tal es el caso del modelo RF, se ha constatado que en la mayoría de los enfoques el valor del hiperparámetro n_estimators es inferior a 50, donde la configuración por defecto establece 100. Por otro lado, las configuraciones identificadas para el modelo SVM también difieren. Para el hiperparámetro C, la mayoría de los enfoques asignaron un valor superior al predeterminado. Además, se obtuvo otro valor para el hiperparámetro *kernel*, el tipo polinomial (grados 2 y 5) arrojan resultados positivos al igual que el tipo rbf, que está establecido de forma predeterminada. Finalmente, en el caso del modelo XGBoost, se encontró un valor menor para el hiperparámetro n_estimators en comparación con el valor establecido por defecto. No obstante, el valor del hiperparámetro max_depth se mantuvo cercano al valor predeterminado en la configuración.

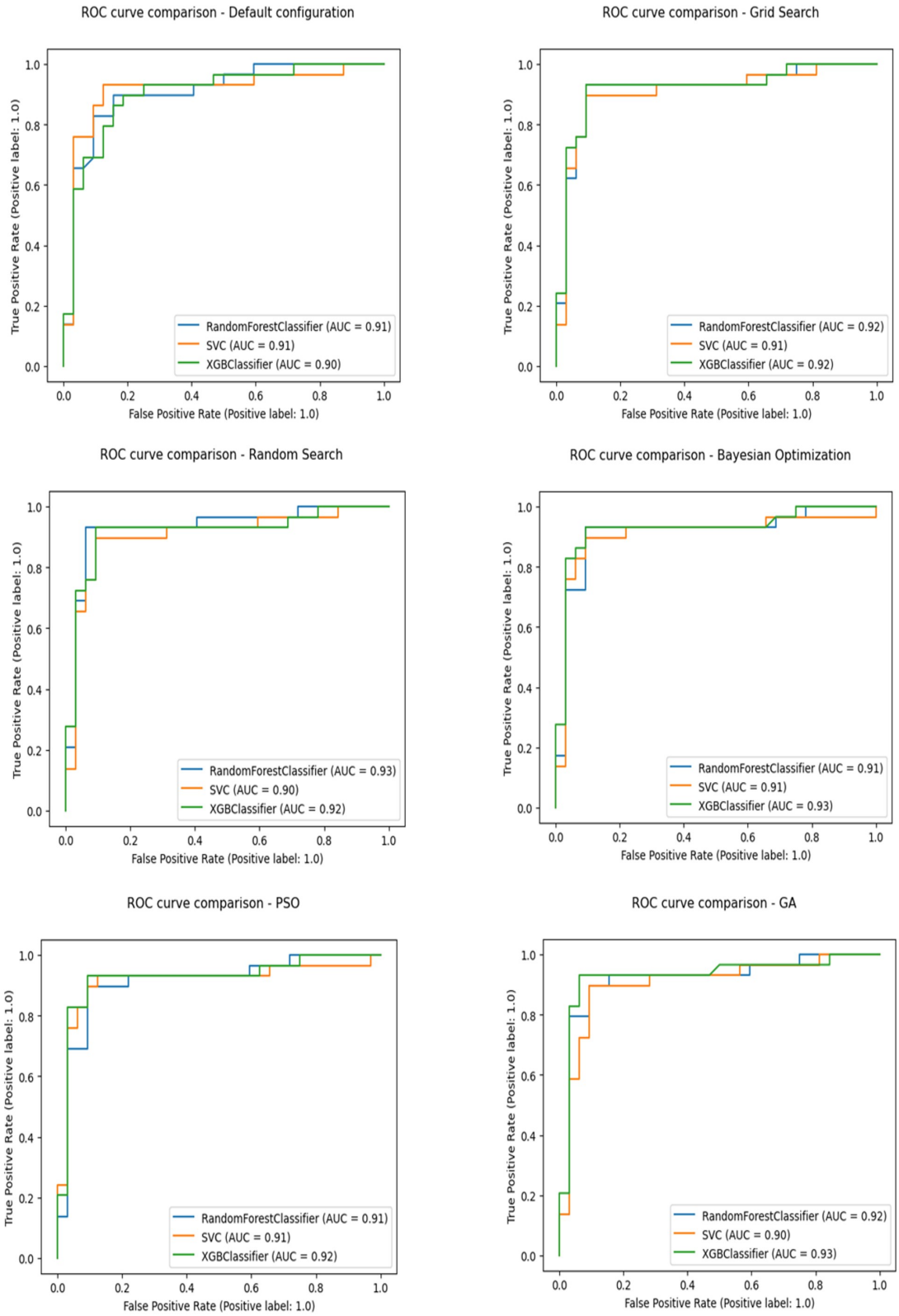


Figura 5.3: Curvas ROC-AUC de los modelos evaluados con el subconjunto de datos de Cleveland.

Algoritmo	Hiperparámetro	GS	RS	BO	PSO	GA
RF	n_estimators	15	25	35	66	43
	max_depth	4	5	7	7	14
	min_samples_leaf	6	5	3	7	11
	min_samples_split	-	4	2	5	9
SVM	c	2.5	0.1	2	24	30.3
	gamma	0.004	0.05	0.02	0.017	0.0001
	kernel	rbf	poly	rbf	rfb	poly
	degree	-	2	-	-	5
	coef0	-	1.5	1	0.15	-0.9
XGBoost	n_estimators	45	15	45	149	36
	learning_rate	0.08	0.3	0.05	0.24	0.3
	gamma	3	1.5	0.6	0.43	0.1
	subsample	0.8	1	0.8	0.9	0.8
	max_depth	5	5	5	4	
	min_child_weight	-	-	4	7.35	7
	colsample_bytree	-	0.8	0.9	0.31	0.6

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.5: Configuración de hiperparámetros relevantes de los modelos evaluados con el subconjunto de datos de Cleveland.

5.1.2 Optimización de hiperparámetros con base en Framingham

En Tabla 5.7 se observa que los cinco enfoques de optimización determinaron combinaciones de hiperparámetros con la capacidad de superar a los modelos con la configuración predeterminada. Los modelos SVM y XGBoost optimizados con el enfoque BO obtuvieron 209 instancias clasificadas correctamente.

El gráfico de barras de la Figura 5.4 muestra que los modelos SVM y XGBoost destacan en la identificación precisa de pacientes con riesgo cardiovascular, al ser optimizados con los enfoques BO y PSO, respectivamente. Por su parte, el modelo RF obtiene mejor desempeño cuando es optimizado con el enfoque PSO.

Los resultados de la evaluación de los modelos considerando todas las métricas se muestran en la Tabla 5.9. El enfoque BO maximiza el porcentaje de rendimiento de los tres modelos a partir de las combinaciones de hiperparámetros.

Algoritmo	Enfoque	Verdaderos Positivos	Falsos Negativos	Falsos Positivos	Verdaderos Negativos	ICC*
RF	Defecto	104	7	15	94	198
	GS	100	11	22	87	187
	RS	103	8	11	98	201
	BO	102	9	8	101	203
	PSO	105	6	12	97	202
	GA	103	8	9	100	203
SVM	Defecto	83	28	24	85	168
	GS	105	6	22	87	192
	RS	107	4	10	99	206
	BO	105	6	5	104	209
	PSO	107	4	10	99	206
	GA	107	4	9	100	207
XGBoost	Defecto	99	12	8	101	200
	GS	100	11	4	105	205
	RS	99	12	5	104	203
	BO	104	7	4	105	209
	PSO	105	6	9	100	205
	GA	104	7	6	103	207

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm, ICC: Instancias Clasificadas Correctamente.

Tabla 5.7: Valores de la matriz de confusión de los modelos evaluados con el subconjunto de datos de Framingham.

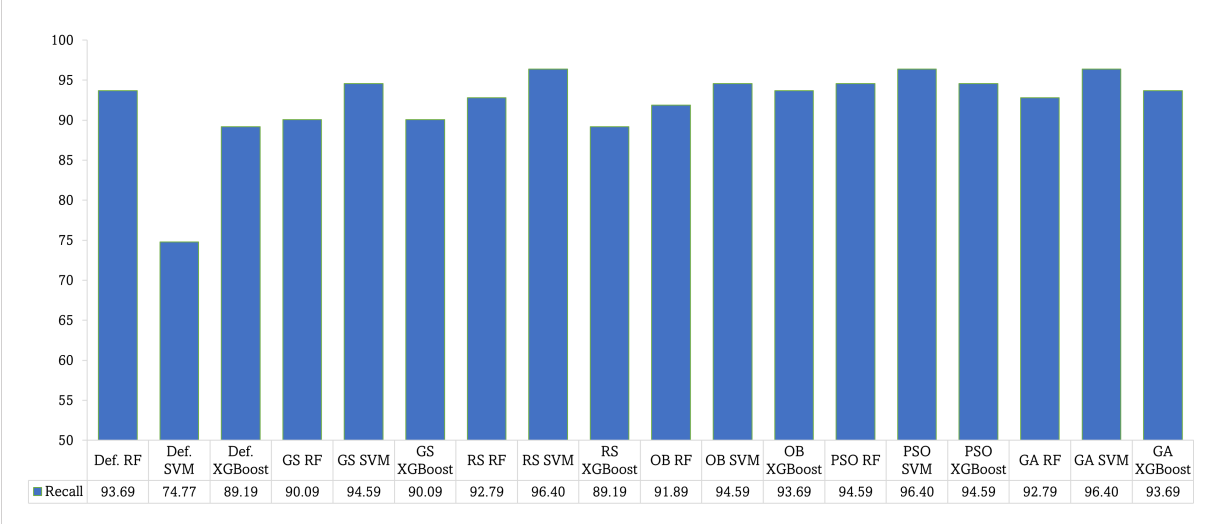


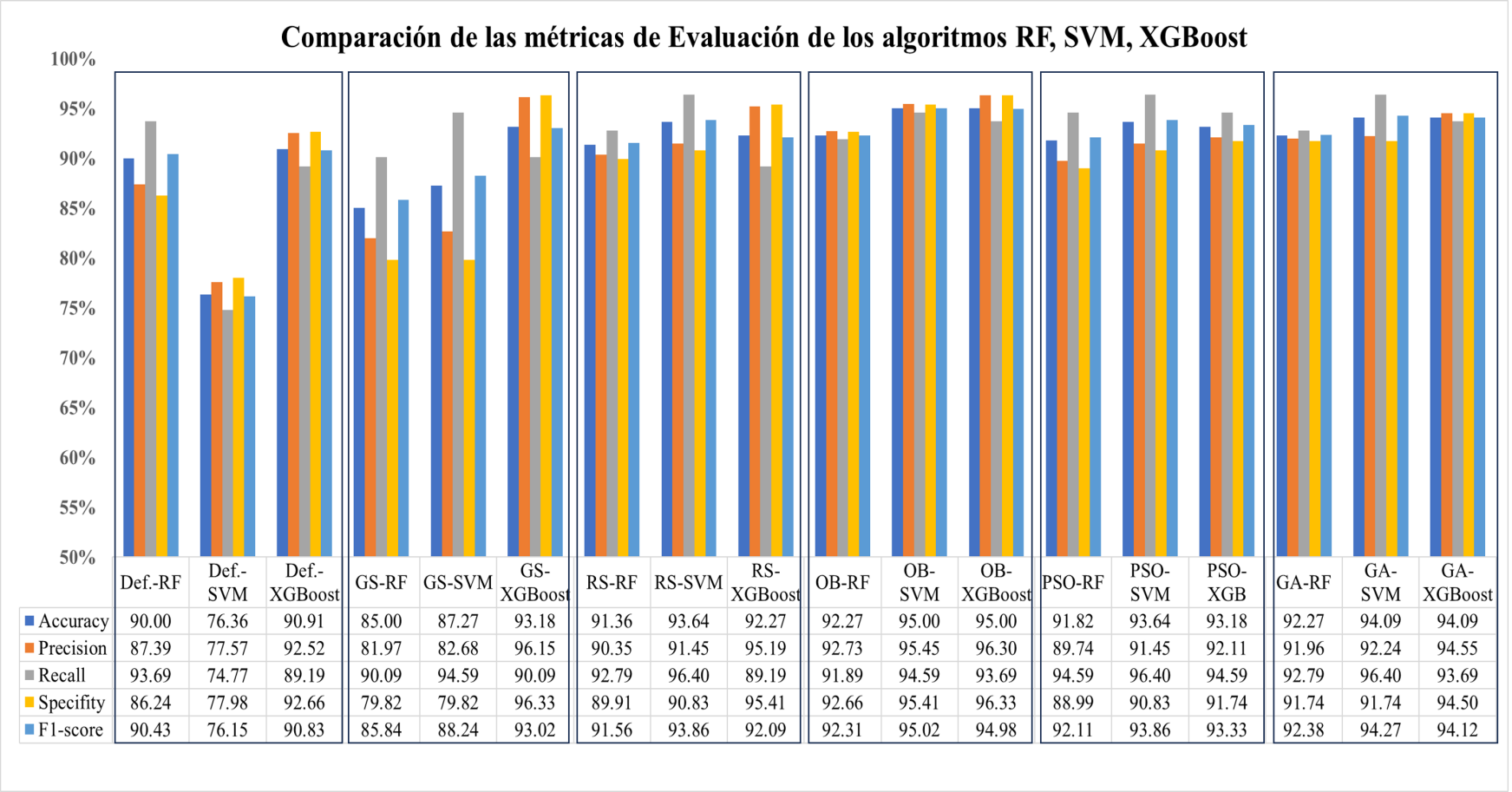
Figura 5.4: Desempeño de los modelos en relación con la métrica *Recall*, evaluados con el subconjunto de datos de Framingham.

Algoritmo	Enfoque	Accuracy train	Accuracy test	Precision	Recall	Specifity	F1-Score
	Defecto	90.15	90.00	87.39	93.69	86.24	90.43
	GS	82.90	85.00	81.97	90.09	79.82	85.84
	RS	88.18	91.36	90.35	92.79	89.91	91.56
RF	BO	89.98	92.27	92.73	91.89	92.66	92.31
	PSO	89.28	91.82	89.74	94.59	88.99	92.11
	GA	88.18	92.27	91.96	92.79	91.74	92.38
	Defecto	73.57	76.36	77.57	74.77	77.98	76.15
	GS	84.74	87.27	82.68	94.59	79.82	88.24
	RS	93.03	93.64	91.45	96.40	90.83	93.86
SVM	BO	92.79	95.00	95.45	94.59	95.41	95.02
	PSO	92.85	93.64	91.45	96.40	90.83	93.86
	GA	92.5	94.09	92.24	96.40	91.74	94.27
	Defecto	90.35	90.91	92.52	89.19	92.66	90.83
	GS	91.08	93.18	96.15	90.09	96.33	93.02
	RS	89.04	92.27	95.19	89.19	95.41	92.09
XGBoost	BO	91.67	95.00	96.30	93.69	96.33	94.98
	PSO	91.34	93.18	92.11	94.59	91.74	93.33
	GA	90.97	94.09	94.55	93.69	94.50	94.12

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.9: Desempeño de los modelos evaluados con el subconjunto de datos de Framingham.

La Figura 5.5 muestra un gráfico de barras del comparativo entre cinco métricas, cada rectángulo vertical hare referencia a un enfoque de optimización. En términos generales, BO es el enfoque de optimización que propone configuraciones para los modelos SVM y XGBoost que maximizan el desempeño de la predicción. Sin embargo, los enfoques PSO y GA también incrementan el desempeño del modelo RF.



	Def.-RF	Def.-SVM	Def.-XGBoost	GS-RF	GS-SVM	GS-XGBoost	RS-RF	RS-SVM	RS-XGBoost	OB-RF	OB-SVM	OB-XGBoost	PSO-RF	PSO-SVM	PSO-XGB	GA-RF	GA-SVM	GA-XGBoost
Accuracy	90.00	76.36	90.91	85.00	87.27	93.18	91.36	93.64	92.27	92.27	95.00	95.00	91.82	93.64	93.18	92.27	94.09	94.09
Precision	87.39	77.57	92.52	81.97	82.68	96.15	90.35	91.45	95.19	92.73	95.45	96.30	89.74	91.45	92.11	91.96	92.24	94.55
Recall	93.69	74.77	89.19	90.09	94.59	90.09	92.79	96.40	89.19	91.89	94.59	93.69	94.59	96.40	94.59	92.79	96.40	93.69
Specifity	86.24	77.98	92.66	79.82	79.82	96.33	89.91	90.83	95.41	92.66	95.41	96.33	88.99	90.83	91.74	91.74	91.74	94.50
F1-score	90.43	76.15	90.83	85.84	88.24	93.02	91.56	93.86	92.09	92.31	95.02	94.98	92.11	93.86	93.33	92.38	94.27	94.12

Figura 5.5: Resultado de las métricas de desempeño de los modelos evaluados con el subconjunto de datos de Framingham.

Por otro lado, al evaluar los modelos con el subconjunto de datos Framingham los gráficos de curva ROC muestran una tendencia positiva, lo que indica que los modelos tienen un excelente rendimiento. En promedio área bajo la curva de los modelos RF, SVM y XGBoost sobrepasa el 0.95, lo que indica que el modelo tiene una buena capacidad de discriminación. Este valor demuestra que nuestros modelos están tomando decisiones informadas y consistentes en un rango de umbrales (ver Figura 5.6).

La Tabla 5.10 muestra las configuraciones de hiperparámetros que brindan a mejor desempeño a los modelos. Cabe mencionar que algunos enfoques obtienen diferentes valores para los hiperparámetros, por ejemplo, cuando se realiza el proceso de optimización del modelo RF mediante los enfoques de optimización GS, PSO y GA, respectivamente, se obtienen diferentes valores para los hiperparámetros n_estimators, max_depth y min_samples_split, Sin embargo, existen enfoques de optimización que convergen sobre un mismo rango de valores, tal es el caso de los enfoques RS y BO, donde el valor de n_estimators es el mismo. Estos resultados ponen de relieve la naturaleza compleja del espacio de hiperparámetros y cómo diferentes soluciones pueden ser igualmente eficaces.

En el modelado del clasificador RF se identificó que la mejora en el rendimiento del modelo está directamente relacionada con el aumento tanto del hiperparámetro n_estimators como de max_depth. Por otro lado, el hiperparámetro C del modelo SVM se ajusta de manera precisa en el rango [1-20] y el tipo de *kernel* a una función polinómica o rbf, con ello se logra obtener un modelo con una mayor capacidad de generalización y precisión en la predicción del riesgo cardiovascular. Por último, aumentar el valor del hiperparámetro n_estimators en el modelo XGBoost puede capturar relaciones más complejas en los datos, lo que puede resultar en un mejor desempeño de las predicciones. Con el ajuste adecuado de los hiperparámetros learning_rate, gamma y max_depth, se pueden controlar el nivel de complejidad del modelo y prevenir el sobreajuste.

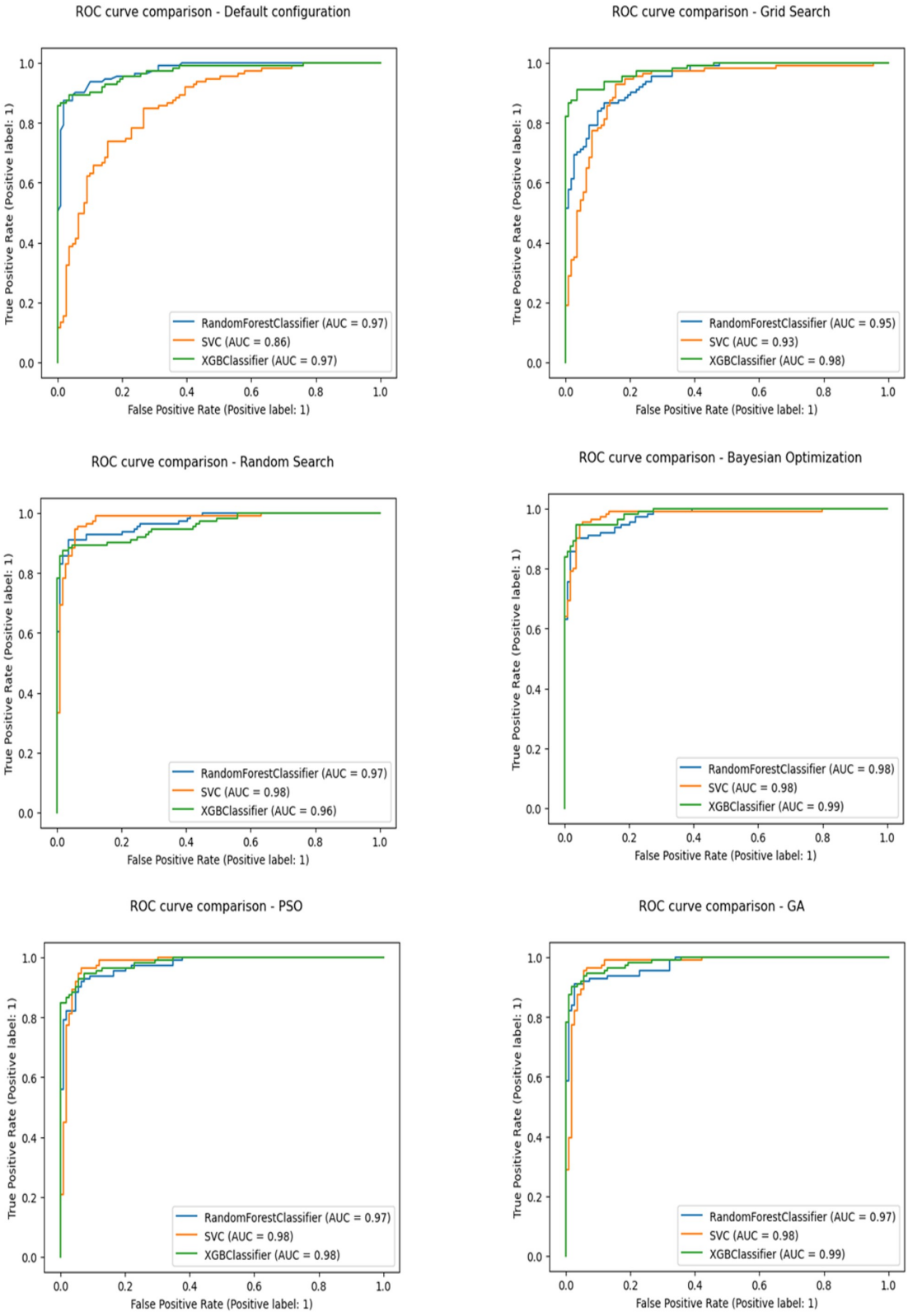


Figura 5.6: Curvas ROC-AUC de los modelos evaluados con el subconjunto de datos de de Framingham.

Algoritmo	Hiperparámetro	GS	RS	BO	PSO	GA
RF	n_estimators	120	200	200	254	90
	max_features	0.9	1	1	1	1
	max_depth	10	30	25	26	35
	min_samples_split	2	10	5	6	4
	min_samples_leaf	1	6	4	1	5
SVM	c	60	3	2.5	19.5	7.7
	gamma	scale	1	1.5	0.1	1
	kernel	rbf	rbf	rbf	rbf	rbf
	coef0	0	0	0	0.10	0.70
	degree	-	-	-	-	-
XGBoost	n_estimators	170	120	200	180	170
	learning_rate	0.1	0.25	0.1	0.13	0.3
	gamma	0	0	0	0.1	0.23
	subsample	0.8	0.8	0.8	0.4	0.8
	max_depth	10	4	17	23	12
	colsample_bytree	0.6	0.6	0.8	0.8	0.9

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.10: Configuración de hiperparámetros relevantes de los modelos evaluados con el subconjunto de datos de Framingham.

5.1.3 Optimización de hiperparámetros con base en Faisalabad

Las configuraciones de los hiperparámetros de los modelos XGBoost y SVM optimizados con los enfoques RS y BO, respectivamente, clasifican más instancias correctamente (79 y 78 de un total de 83). Cabe mencionar que el número máximo de instancias correctamente clasificadas utilizando la configuración de hiperparámetros fue 75.

El modelo RF obtuvo el 100 % en la identificación correcta de paciente que sufrieron evento de muerte a causa de insuficiencia cardiaca, con todos los enfoques de optimización; el modelo SVM obtuvo 100 % con los enfoques GS y BO, con resto de los enfoques un 97.56 % (ver Figura 5.7).

Algoritmo	Enfoque	Verdaderos Negativos	Falsos Positivos	Falsos Negativos	Verdaderos Positivos	ICC*
	Defecto	35	7	1	40	75
	GS	36	6	0	41	77
	RS	36	6	0	41	77
RF	BO	37	5	0	41	78
	PSO	36	6	0	41	77
	GA	35	7	0	41	76
	Defecto	31	11	0	41	72
	GS	36	6	0	41	77
SVM	RS	36	6	1	40	76
	BO	36	6	0	41	77
	PSO	37	5	1	40	77
	GA	37	5	1	40	77
	Defecto	36	6	4	37	73
	GS	36	6	1	40	76
XGBoost	RS	38	4	0	41	79
	BO	37	5	1	40	77
	PSO	35	7	0	40	77
	GA	35	7	0	41	76

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm, ICC: Instancias Clasificadas Correctamente.

Tabla 5.12: Valores de la matriz de confusión de los modelos evaluados con el subconjunto de datos de Faisalabad.

Se logró definir una combinación de valores para los hiperparámetros a través de los cinco enfoques de optimización, lo que otorgó a los modelos la habilidad de detectar a los pacientes que fallecieron a causa de insuficiencia cardíaca durante el estudio realizado (métrica *Recall*). Aunque la cantidad de registros en el conjunto de datos de Faisalabad es similar a Cleveland, se destacan diferencias notables. Se obtuvo un valor superior para el hiperparámetro n_estimators en el modelo RF cuando se emplearon los enfoques GS y BO. Además, este valor excede el establecido en la configuración por defecto. Ocurre el mismo patrón con el hiperparámetro max_depth. En la Tabla 5.14 se presentan los resultados de la evaluación de los modelos generados.

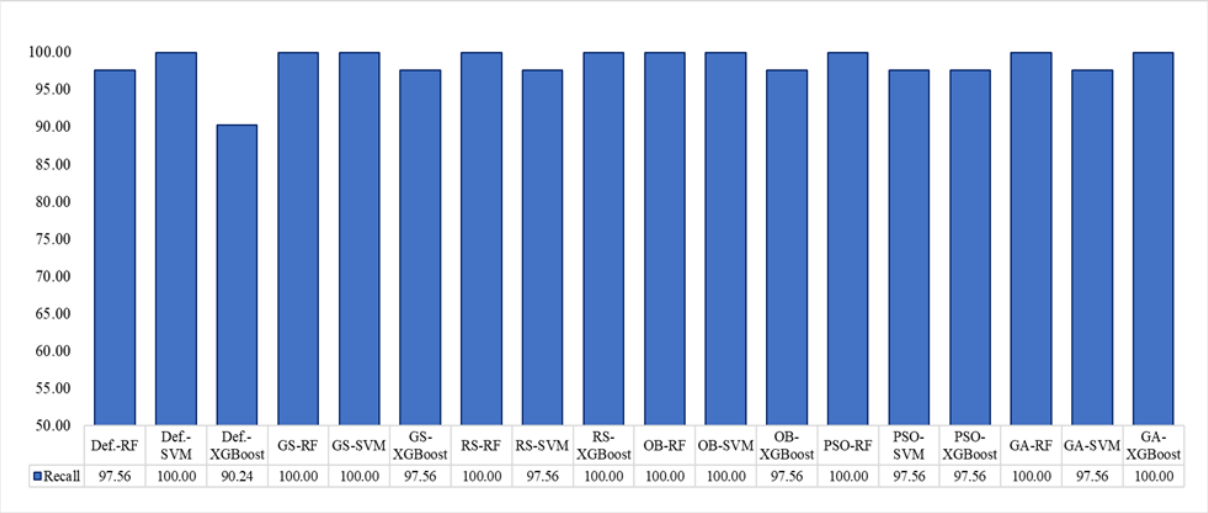


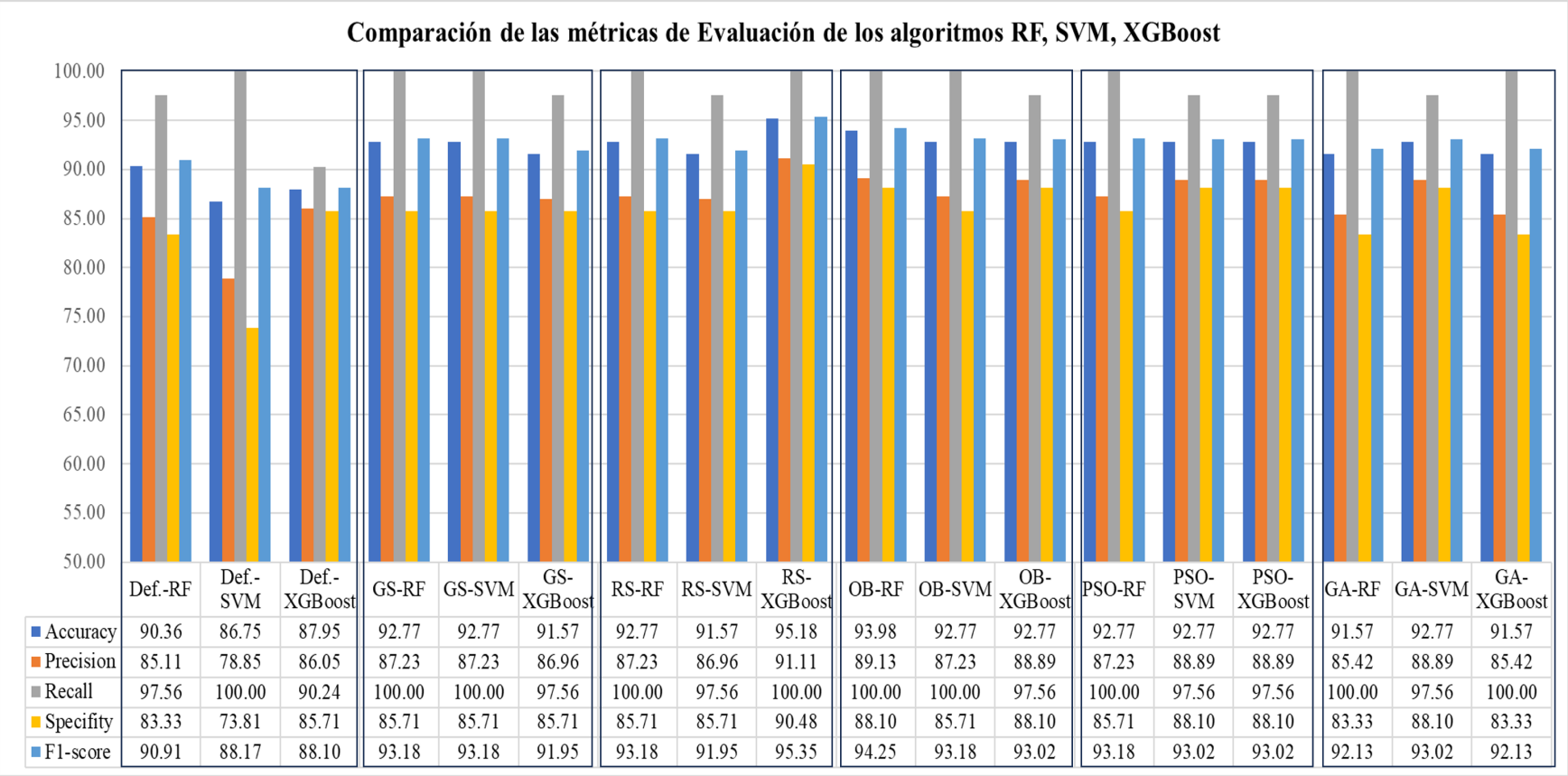
Figura 5.7: Desempeño de los modelos en relación con la métrica *Recall*, evaluados con el subconjunto de datos de Faisalabad.

Algoritmo	Enfoque	Accuracy train	Accuracy test	Precision	Recall	Specifity	F1-Score
	Defecto	87.76	90.36	85.11	97.56	83.33	90.91
	GS	89.06	92.77	87.23	100	85.71	93.18
	RS	85.97	92.77	87.23	100	85.71	93.18
RF	BO	87.17	93.98	89.13	100	88.10	94.25
	PSO	83.86	92.77	87.23	100	85.71	93.18
	GA	85.05	91.57	85.42	100	83.33	92.13
	Defecto	83.87	86.75	78.85	100	73.81	88.17
	GS	83.24	92.77	87.23	100	85.71	93.18
	RS	83.85	91.57	86.96	97.56	85.71	91.95
SVM	BO	88.08	92.77	87.23	100	85.71	93.18
	PSO	84.44	92.77	88.89	97.56	88.10	93.02
	GA	81.71	92.77	88.89	97.56	88.10	93.02
	Defecto	86.91	87.95	86.05	90.24	85.71	88.10
	GS	86.62	91.57	86.96	97.56	85.71	91.95
XGBoost	RS	86.92	95.18	91.11	100	90.48	95.35
	BO	85.93	92.77	88.89	97.56	88.10	93.02
	PSO	87.77	92.77	88.89	97.56	88.10	93.02
	GA	88.12	91.57	85.42	100	83.33	92.13

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.14: Desempeño de los modelos evaluados con el subconjunto de datos de Faisalabad.

La Figura 5.8 muestra un gráfico de barras del comparativo entre cinco métricas, cada rectángulo vertical hare referencia a un enfoque de optimización. La configuración de hiperparámetros del modelo XGBoost determinada por el enfoque de optimización RS presenta el mejor modelo de predicción, en términos de las métricas *accuracy y F1 score*. Por su parte, el enfoque de optimización PSO brinda la mayor capacidad de predicción de los registros con evento de muerte a causa de insuficiencia cardiaca en los modelos RF y XGBoost.



	Def.-RF	Def.-SVM	Def.-XGBoost	GS-RF	GS-SVM	GS-XGBoost	RS-RF	RS-SVM	RS-XGBoost	OB-RF	OB-SVM	OB-XGBoost	PSO-RF	PSO-SVM	PSO-XGBoost	GA-RF	GA-SVM	GA-XGBoost
Accuracy	90.36	86.75	87.95	92.77	92.77	91.57	92.77	91.57	95.18	93.98	92.77	92.77	92.77	92.77	92.77	91.57	92.77	91.57
Precision	85.11	78.85	86.05	87.23	87.23	86.96	87.23	86.96	91.11	89.13	87.23	88.89	87.23	88.89	88.89	85.42	88.89	85.42
Recall	97.56	100.00	90.24	100.00	100.00	97.56	100.00	97.56	100.00	100.00	100.00	97.56	100.00	97.56	97.56	100.00	97.56	100.00
Specifity	83.33	73.81	85.71	85.71	85.71	85.71	85.71	85.71	90.48	88.10	85.71	88.10	85.71	88.10	88.10	83.33	88.10	83.33
F1-score	90.91	88.17	88.10	93.18	93.18	91.95	93.18	91.95	95.35	94.25	93.18	93.02	93.18	93.02	93.02	92.13	93.02	92.13

Figura 5.8: Resultado de las métricas de desempeño de los modelos evaluados con el subconjunto de datos de Faisalabad.

El gráfico de las curvas ROC muestran una tendencia positiva, lo que indica que los modelos tienen un rendimiento mucho mejor que al azar en la mayoría de los umbrales. El área bajo la curva de los modelos RF, SVM y XGBoost sobrepasa el 0.93, lo que demuestra que los modelos tienen una buena capacidad de discriminación. En la Figura 5.9 se muestran los gráficos ROC-AUC de los modelos entrenados en cada uno de los enfoques de optimización de hiperparámetros.

Finalmente, la Tabla 5.15 muestra las configuraciones de hiperparámetros que mejoran la capacidad predictiva de los modelos. Es importante destacar que el valor de algunos hiperparámetros difiere entre los enfoques de optimización. Podemos observar que las combinaciones determinadas por los enfoques RS, PSO y GA proporcionan al modelo RF valores del hiperparámetro n_estimators por debajo del valor establecido por defecto. Para el caso del modelo SVM, todos los enfoques optimización determinaron que el mejor valor para el hiperparámetro kernel es: poly y rbf, con valores de penalización pequeños (hiperparámetro c). Por su parte, en el modelo XGBoost, el valor del hiperparámetro n_estimators obtenido por todos los enfoques de optimización es menor e igual al establecido por defecto. Además, los enfoques GS y GA determinan el mismo valor para el hiperparámetro learning_rate.

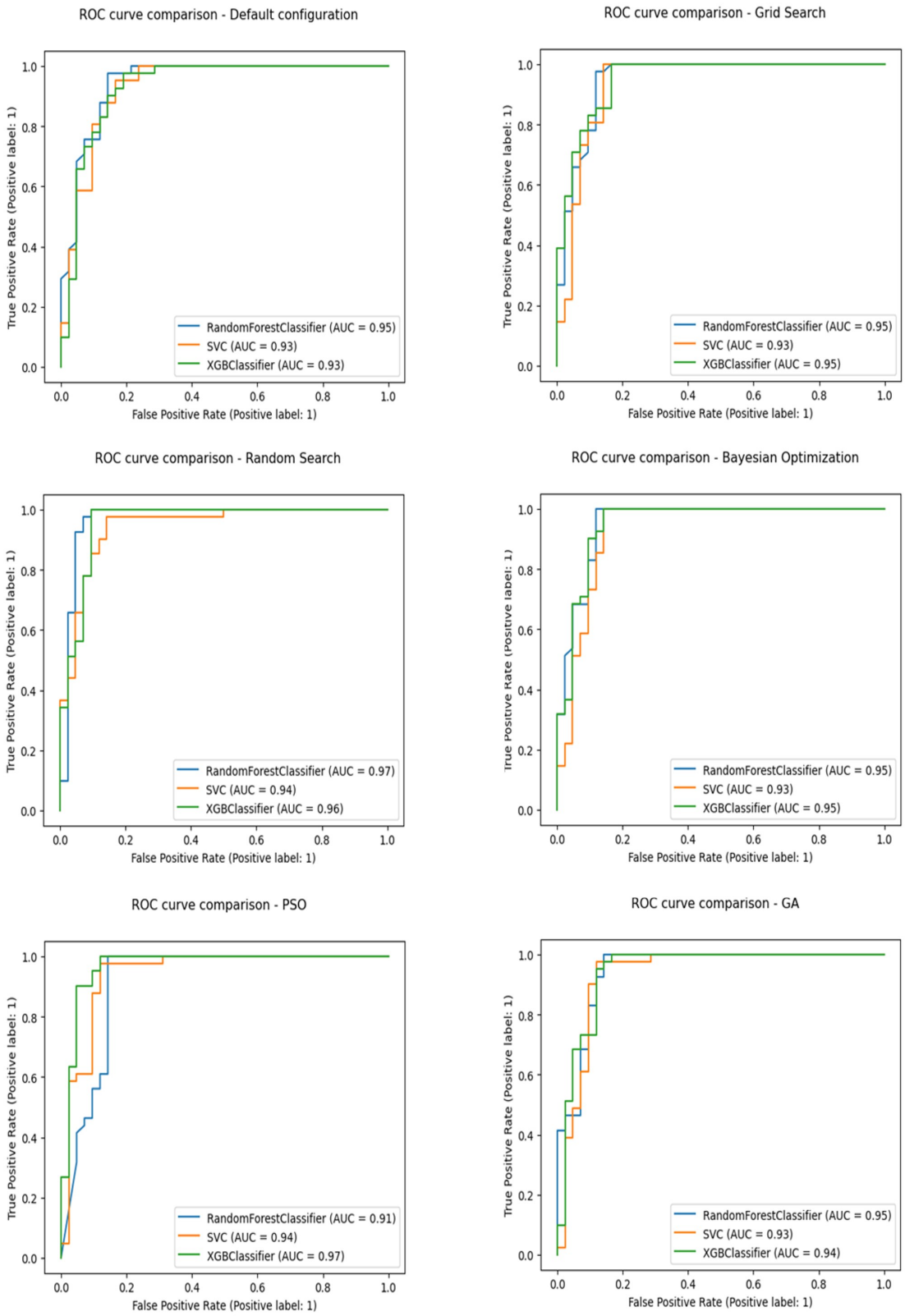


Figura 5.9: Curvas ROC-AUC de los modelos evaluados con el subconjunto de datos de Faisalabad.

Algoritmo	Hiperparámetro	GS	RS	BO	PSO	GA
RF	n_estimators	130	45	140	51	44
	max_features	'auto'	1	1	0.7	1
	max_depth	12	7	'none'	12	11
	min_samples_split	-	6	1	3	7
	min_samples_leaf	-	2	2	2	4
SVM	c	7	3.5	5.0	1.5	25
	gamma	-	-	0.1	0.84	0.2
	kernel	poly	poly	rbf	rbf	rbf
	degree	3	3	-	-	-
XGBoost	n_estimators	40	100	35	60	89
	learning_rate	0.12	0.4	0.6	0.1	0.12
	gamma	-	-	-	-	0.2
	subsample	0.7	1	0.7	0.9	0.6
	max_depth	7	4	20 6	12	
	min_child_weight	-	-	-	-	1
	colsample_bytree	0.7	0.8	0.8	0.6	0.8

GS: Grid Search, RS: Random Search, BO: Bayesian Optimization, PSO: Particle Swarm Optimization, GA: Genetic Algorithm

Tabla 5.15: Configuración de hiperparámetros relevantes de los modelos evaluados con el subconjunto de datos de Faisalabad.

5.2. Enfoques de optimización relevantes

Esta investigación mostró cómo los enfoques de optimización pueden maximizar el desempeño de los modelos de clasificación, identificando combinaciones de hiperparámetros prometedoras que mejoran la precisión y otras medidas de desempeño. La selección del enfoque de optimización debe tener en cuenta la complejidad de los hiperparámetros que contiene el modelo y las particularidades de los datos, ya que cada estrategia posee mecanismos de búsqueda diferentes y por tanto proporciona resultados diferentes.

La Tabla 5.17 muestra los enfoques de optimización que determinaron la mejor configuración de hiperparámetros para cada algoritmo de aprendizaje automático en particular. Los enfoques BO, PSO y GA proporcionaron mejoras significativas en las métricas: *accuracy, recall, specificity*. Sin embargo, se detectaron algunas debilidades con respecto a la definición de los espacios de búsqueda de los hiperparámetros de tipo discreto (son aquellos que toman valores

Conjunto de datos	Modelo	Enfoque de Optimización	Instancias Clasificadas	Recall
Cleveland	RF	*Random Search*	57 de 61	93.10
	SVM	*Grid Search*	55 de 61	89.66
	XGBoost	*Random Search*	56 de 61	93.75
Framingham	RF	*Particle Swarm*	202 de 220	94.59
	SVM	*Genetic Algorithm*	207 de 220	96.40
	XGBoost	*Particle Swarm*	205 de 220	94.59
Faisalabad	RF	*Bayesian*	78 de 83	100
	SVM	*Grid Search, Bayesian*	77, 78 de 83	93.18
	XGBoost	*Random Search*	79 de 83	100

Tabla 5.17: Enfoques de optimización relevantes en cada conjunto de datos.

específicos y separados, en contraste con los hiperparámetros continuos que pueden tomar un rango de valores dentro de un intervalo). El enfoque BO cuando optimiza una función, asume que los hiperparámetros de entrada son continuos ya que este método utiliza una función de adquisición que se define solo en un dominio continuo. En el proceso de optimización de hiperparámetros es común encontrar variables discretas, como el número de árboles y la profundidad de los árboles cuando se ajusta un modelo RF. Por lo tanto, aplicar BO para optimizar funciones con entradas discretas es un problema desafiante.

Sin embargo, un problema clave en la aplicación de los enfoques PSO y GA es el amplio espacio de búsqueda en el que residen los hiperparámetros, lo que puede llevar a dificultades en la exploración efectiva. Además, existe un equilibrio delicado entre la exploración de nuevas regiones y la explotación de soluciones prometedoras, y tanto PSO como GA pueden inclinarse hacia una de estas estrategias, lo que podría comprometer la calidad de la búsqueda y conducir a convergencia prematura. Otro desafío es la necesidad de ajustar de manera precisa sus parámetros propios, así como de definir la métrica de optimización adecuada. La configuración de estos algoritmos puede ser compleja y crítica para su rendimiento. Además, su sensibilidad a la configuración podría afectar la calidad de los resultados obtenidos. Asimismo, estos mecanismos pueden requerir un tiempo computacional considerable, lo que podría ser una preocupación en aplicaciones donde la eficiencia temporal es crucial.

5.3. Comparación con el estado del arte

La Tabla 5.19 muestra un estudio comparativo entre los modelos obtenidos en esta investigación y los reportados en la literatura.

ID	Autores	Modelo - Enfoque	Conjunto de datos	% *accuracy*
1	Guarneros-Nolasco et al., 2021	SVM	Cleveland	82.42
		RF	Framingham	83.01
		DT	Faisalabad	76.59
		LR	South African Hearth	73.38
2	Reddy et al., 2021	SMO	Cleveland	85.14
3	Shah et al., 2020	k-NN	Cleveland	90.78
4	Allah et al., 2022	XGBoost	Cleveland	91.6
			Kagge	100
5	Gupta et al., 2020	LR	Cleveland	93.44
6	Ayon et al., 2020	SVM	Cleveland	97.36
		DNN	Statlog	98.15
7	Li et al., 2020	SVM	Cleveland	92.37
8	Asadi et al., 2021	RF - MOPSO	Cleveland	85.21
			Statlog	88.26
			SPECT	87.65
			SPECTF	86.70
			Long Beach VA	87.50
9	Rohit et al., 2021	ANN - GS	Cleveland/Hungary/ Switzerland/Long Beach V.	94.20
10	Ghosh et al., 2021	RFBM - GS	Cleveland/Long Beach V. /Switzerland/Hungary /Statlog	99.05
11	Valarmathi y Sheela, 2021	RF - TPOT	Cleveland	97.52
		RF - RS	Z-Alizadeh Sani	80.20
12	Kabir y Zaman, 2020	k-NN - GS	Cleveland	91.80
13	Budholiya et al., 2022	XGBoost - BO	Cleveland	91.80
14	**TESIS, 2023**	**RF - RS**	**Cleveland**	**93.44**
		(SVM, XGBoost) - BO	**Framingham**	**95.00**
		XGBoost - RS	**Faisalabad**	**95.18**

Tabla 5.19: Comparación de esta investigación con los resultados reportados en la literatura

Los modelos seleccionados de la literatura aplican, en su mayoría, los mecanismos de optimización de hiperparámetros considerados en esta investigación: predeterminado, manual o mediante enfoques de optimización. Los trabajos del estado del arte mencionados en este estudio comparativo son de los últimos cinco años.

Se resumen distintos enfoques de modelos de aprendizaje automático aplicados a diferentes conjuntos de datos médicos para predicción de enfermedades cardíacas. Se observa que diferentes autores han empleado diversos modelos, como SVM, RF, DT, LR, k-NN, XGBoost, ANN, y otros. El porcentaje de *accuracy* varía desde alrededor del 73 % hasta el 100 %, dependiendo del modelo y el conjunto de datos. Además, se destaca que nuestra propuesta que utiliza los modelos RF, SVM, XGBoost en tres conjuntos de datos distintos, con resultados de *accuracy* que oscilan entre el 93.44 % y el 95.18 %.

Capítulo 6

Conclusiones

En este proyecto de investigación se optimizó el desempeño de los modelos de predicción de enfermedades cardiovasculares utilizando los algoritmos de aprendizaje automático *Random Forest, Support Vector Machine y XGBoost*. Estos modelos se construyeron utilizando los conjuntos de datos Cleveland, Framingham y Faisalabad, tomados de repositorios públicos. Se cumplió con el objetivo el cual consistió en determinar la configuración de hiperparámetros que maximiza el desempeño de los modelos de predicción, tomando como referencia el desempeño obtenido con la configuración predeterminada de los hiperparámetros. Para lograr esto, se aplicaron los enfoques de optimización *Grid Search, Random Search, Bayesian Optimization, Particle Swarm Optimization y Genetic Algorithm*.

Las proporciones de registros para la fase de entrenamiento y prueba se realizaron conforme los tamaños de cada conjunto de datos. En el caso de Cleveland, cuenta con pocos registros. Por ello, se optó por una proporción del 80 % para entrenamiento y 20 % para prueba, asegurando un equilibrio de los datos que se necesitan para entrenar el modelo y para evaluar su generalización. Por otro lado, el conjunto de datos Framingham, con una mayor cantidad de registros, se dividió en 97 % para entrenamiento y 3 % para prueba, permitiendo aprovechar los datos en el entrenamiento y mantener un conjunto con instancias suficientes para la fase de evaluación. Por último, el conjunto de datos Faisalabad, con un tamaño parecido al conjunto de datos Cleveland, se consideró una proporción del 80 % para entrenamiento y 20 % para prueba. Esta estrategia de partición se enfocó en obtener modelos optimizados para cada conjunto de datos y maximizar su rendimiento en cada escenario específico.

Los resultados del desempeño por cada métrica destacan los puntos fuertes y débiles de cada modelo, optimizados con distintos enfoques. En general, el modelo SVM destaca su desempeño en términos de las métricas *accuracy y specificity*, demostrando su fiabilidad a la hora de categorizar correctamente tanto instancias positivas como negativas. Por otro lado, XGBoost destaca en la métrica de Recall, demostrando su destreza en la identificación precisa de instancias positivas.

6.1. Productos derivados de la investigación

6.1.1 Actículos de congresos nacionales

1. Sánchez-Jiménez, E., Hernández, Y., Ortiz, J. (2022). Breve revisión de la literatura sobre modelos predictivos para la detección de enfermedades cardiovasculares, *Memorias de la Jornada de Ciencia y Tecnología Aplicada (JCyTA)*, vol. 5, Cuernavaca, Morelos, 2022, pp. 61–66. url: https://jcyta.cenidet.tecnm.mx/revistas/jcyta/08-Revista_JCyTA_Vol-5-Num-1_Ene-Jun_2022.pdf.

2. Sánchez-Jiménez, E., Hernández, Y., Ortiz, J., (2022). Técnicas de optimización de hiperparámetros en modelos de aprendizaje automático para predicción de enfermedades cardiovasculares, *Memorias de la Jornada de Ciencia y Tecnología Aplicada (JCyTA)*, vol. 5, Cuernavaca, Morelos, 2022, pp. 82–88. url: https://jcyta.cenidet.tecnm.mx/revistas/jcyta/09-Revista_JCyTA_Vol-5-Num-2_Jul-Dic_2022.pdf.

3. Sánchez-Jiménez, E., Hernández, Y., Ortiz-Hernández, J., Martínez-Rebollar, A., Estrada-Esquivel, H. (2023). Configuración de hiperparámetros mediante algoritmos de optimización: Aplicación en la predicción de enfermedades cardiovasculares, *Research in Computing Science*, vol. 152, no. 8, 2023, ISSN: 1870-4069. url: https://www.rcs.cic.ipn.mx/2023_152_8/

4. Cuevas-Chávez, A., Sánchez-Jiménez, E., Hernández, Y., Ortiz-Hernández, (2023). Class Balancing and Hyperparameter Configuration for Cardiovascular Disease Prediction. *2023 IEEE Mexican International Conference on Computer Science (ENC).*

6.1.2 Actículos en revistas internacionales

1. Cuevas-Chávez, A., Hernández, Y., Ortiz-Hernandez, J., Sánchez-Jiménez, E., Ochoa-Ruiz, G., Pérez, J., González-Serna, G. (2023). A Systematic Review of Machine Learning and IoT Applied to the Prediction and Monitoring of Cardiovascular Diseases. *Healthcare*, 11, 2240. https://doi.org/10.3390/healthcare11162240.

2. Cuevas-Chávez, A., Narciso, S., Sánchez-Jiménez, E., Celerino-Pérez, I., Hernández, Y.; Ortiz-Hernandez, J. (2023). School dropout prediction with class balancing and hyper-parameter configuration, *Lecture Notes in Artificial Intelligence (LNAI) - Springer*, vol. 14502. https://doi.org/10.1007/978-3-031-51940-6_2.

3. Sánchez-Jiménez, E., Cuevas-Chávez, A., Hernández, Y., Ortiz-Hernández, J., Hernández, José-Alberto, Martínez-Rebollar, A., Estrada-Esquivel, H. (2023). Hyperparameter optimization approaches to improve the performance of machine learning models for cardiovascular risk prediction, *Journal of Intelligent Fuzzy Systems (JIFS)*.

6.2. Trabajo futuro

Como trabajo futuro se identificó un área de oportunidad para implementar/aplicar mecanismos que permitan abordar la configuración efectiva de los parámetros que pertenecen a las metaheurísticas PSO y GA. La optimización de los parámetros internos de las metaheurísticas, como la velocidad de las partículas en PSO o la tasa de cruce y la tasa de mutación en GA, se presenta como un desafío esencial en la aplicación de estas técnicas a problemas específicos. La sensibilidad de estas configuraciones a menudo exige ajustes personalizados para cada problema, lo que refleja la adaptación necesaria al contexto particular. A medida que la búsqueda avanza y la población de soluciones evoluciona, la dinámica cambia, lo que hace necesario modificar los parámetros internos en consecuencia (Bischl et al., 2023). La investigación constante en el campo de las metaheurísticas conduce al desarrollo de nuevas variantes y estrategias para optimizar estos ajustes.

Glosario

- Algoritmo Genético, del inglés *Genetic Algorithm*. Algoritmo de optimización que se basa en principios de evolución biológica, como la selección natural, la mutación y la recombinación, para encontrar soluciones óptimas.
- Búsqueda Aleatoria, del inglés *Random search*. Enfoque de optimización de hiperparámetros que selecciona configuraciones de hiperparámetros de manera aleatoria dentro de un rango predefinido.
- Búsqueda en Rejilla, del inglés *Grid search*. Técnica de optimización de hiperparámetros que explora sistemáticamente combinaciones predefinidas de valores de hiperparámetros para encontrar la configuración óptima.
- Función de costo/pérdida. Función que mide la discrepancia entre las predicciones del modelo y los valores reales.
- Optimización Bayesiana, del inglés *Bayesian Optimization*. Método de optimización de hiperparámetros que utiliza modelos probabilísticos para guiar la búsqueda hacia las configuraciones óptimas.
- Optimización por Enjambre de Partículas, del inglés *Particle Swarm Optimization*. Algoritmo de optimización inspirado en el comportamiento de enjambres de partículas que busca encontrar soluciones óptimas en un espacio de búsqueda.
- Validación Cruzada, del inglés *Cross Validation*. Técnica que se utiliza para evaluar el rendimiento del modelo mediante la partición del conjunto de datos en subconjuntos de entrenamiento y prueba.

Símbolos matemáticos

- X: Conjunto de características o variables predictoras $X = \{X_1, X_2, \ldots, X_p\}$, donde X_i representa la i-ésima característica.
- y: Variable objetivo que representa la presencia o ausencia de riesgo cardiovascular. $y \in \{0, 1\}$, donde 0 indica ausencia y 1 indica presencia.
- θ: Vector de hiperparámetros del modelo de aprendizaje automático. $\theta = (\theta_1, \theta_2, \ldots, \theta_k)$.
- $\mathcal{D}_{train}$: Conjunto de datos de entrenamiento. $\mathcal{D} = \{(X_i, y_i)\}_{i=1}^{n}$, donde (X_i, y_i) son muestras de entrenamiento.
- $f()$: Función objetivo que se busca optimizar. Representa la medida de rendimiento o la función de pérdida del modelo en términos de los hiperparámetros θ.

Referencias

Abohelwa, M., Kopel, J., Shurmur, S., Ansari, M. M., Awasthi, Y., & Awasthi, S. (2023). The Framingham Study on Cardiovascular Disease Risk and Stress-Defenses: A Historical Review. *Journal of Vascular Diseases*, *2*(1), 122-164. https://doi.org/10.3390/jvd2010010

Aguilar-Salinas, C. A., Rojas, R., Gómez-Pérez, F., Valles, V., Franco, A., Olaiz, G., Tapia-Conyer, R., J, S., & JA, R. (2022). Características de los casos con dislipidemias mixtas en un estudio de población: Resultados de la Encuesta Nacional de Enfermedades Crónicas. *Salud Publica Mex*, *44*. https://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=17059

Allah, E., El-Matary, D., Eid, E., & Dien, A. (2022). Performance Comparison of Various Machine Learning Approaches to Identify the Best One in Predicting Heart Disease. *Journal of Computer and Communications*, *10*(6), 1-18. https://doi.org/https://doi.org/10.4236/jcc.2022.102001

Andonie, R. (2019). Hyperparameter optimization in learning systems. *Journal of Membrane Computing*, *1*, 279-291. https://api.semanticscholar.org/CorpusID:208127837

Asadi, S., Roshan, S., & Kattan, M. W. (2021). Random forest swarm optimization-based for heart diseases diagnosis. *Journal of Biomedical Informatics*, *115*, 103690. https://doi.org/https://doi.org/10.1016/j.jbi.2021.103690

Ayon, S. I., Islam, M. M., & Hossain, M. R. (2020). Coronary Artery Heart Disease Prediction: A Comparative Study of Computational Intelligence Techniques. *IETE Journal of Research*, *68*(4), 2488-2507. https://doi.org/10.1080/03772063.2020.1713916

Benhar, H., Idri, A., & Fernández-Alemán, J. (2020). Data preprocessing for heart disease classification: A systematic literature review. *Computer Methods and Programs in Biomedicine*, *195*, 105635. https://doi.org/https://doi.org/10.1016/j.cmpb.2020.105635

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, *13*(10), 281-305. http://jmlr.org/papers/v13/bergstra12a.html

Birjmohun, R. S., Hutten, B. A., Kastelein, J. J. P., & Stroes, E. S. G. (2005). Efficacy and Safety of High-density Lipoprotein Cholesterol-increasing Compounds: a Meta-analysis of Randomized Controlled Trials. *Journal of the American College of Cardiology*, *45*(2). https://doi.org/10.1016/j.jacc.2004.10.031

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., & Lindauer, M. (2023). Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *WIREs Data Mining and Knowledge Discovery*, *13*(2), e1484. https://doi.org/https://doi.org/10.1002/widm.1484

Bradley, S. P., Hax, A. C., & Magnanti, T. L. (1977). Applied Mathematical Programming. https://api.semanticscholar.org/CorpusID:59791415

British-Heart-Foundation. (2017). *Heart rhythms* [Fecha de acceso: 27 abril de 2022]. https://www.bhf.org.uk/informationsupport/publications/heart-conditions/heart-rhythms

Brochu, E., Cora, V. M., & de Freitas, N. (2010). A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. *CoRR*, *abs/1012.2599*. http://arxiv.org/abs/1012.2599

Budholiya, K., Shrivastava, S. K., & Sharma, V. (2022). An optimized XGBoost based diagnostic system for effective prediction of heart disease. *Journal of King Saud University - Computer and Information Sciences*, *34*(7), 4514-4523. https://doi.org/https://doi.org/10.1016/j.jksuci.2020.10.013

Cervantes, C. A. D. (2019). Tendencia e impacto de la mortalidad por enfermedades cardiovasculares en México, 1990-2015. *Rev Cubana Salud Publica*, *45*. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-34662019000400006

Chang, V., Bhavani, V. R., Xu, A. Q., & Hossain, M. (2022). An artificial intelligence model for heart disease detection using machine learning algorithms. *Healthcare Analytics*, *2*, 100016. https://doi.org/https://doi.org/10.1016/j.health.2022.100016

Chen, T., Keravnou-Papailiou, E., & Antoniou, G. (2021). Medical analytics for healthcare intelligence – Recent advances and future directions. *Artificial Intelligence in Medicine*, *112*, 102009. https://doi.org/https://doi.org/10.1016/j.artmed.2021.102009

El-Hashash, E. F., & Shiekh, R. H. A. (2022). A Comparison of the Pearson, Spearman Rank and Kendall Tau Correlation Coefficients Using Quantitative Variables. *Asian Journal of Probability and Statistics*. https://api.semanticscholar.org/CorpusID:253016162

Espinosa-Paredes, G., & Rodríguez, Á. V. (2016). Aplicaciones de programación no lineal. https://api.semanticscholar.org/CorpusID:183090386

Fernández-Solá, J. (2005). Consumo de alcohol y riesgo cardiovascular. *Hipertensión*, *22*(3), 117-132. https://doi.org/https://doi.org/10.1016/S0212-8241(05)71551-6

Ghosh, P., Azam, S., Jonkman, M., Karim, A., Shamrat, F. M. J. M., Ignatious, E., Shultana, S., Beeravolu, A. R., & De Boer, F. (2021). Efficient Prediction of Cardiovascular Disease Using Machine Learning Algorithms With Relief and LASSO Feature Selection Techniques. *IEEE Access*, *9*, 19304-19326. https://doi.org/10.1109/ACCESS.2021.3053759

Guarneros-Nolasco, L. R., Cruz-Ramos, N. A., Alor-Hernández, G., Rodríguez-Mazahua, L., & Sánchez-Cervantes, J. L. (2021). Identifying the Main Risk Factors for Cardiovascular Diseases Prediction Using Machine Learning Algorithms. *Mathematics*, *9*(20). https://doi.org/10.3390/math9202537

Gupta, A., Kumar, R., Singh Arora, H., & Raman, B. (2020). MIFH: A Machine Intelligence Framework for Heart Disease Diagnosis. *IEEE Access*, *8*, 14659-14674. https://doi.org/10.1109/ACCESS.2019.2962755

Hazan, E., Klivans, A. R., & Yuan, Y. (2017). Hyperparameter Optimization: A Spectral Approach. *CoRR*, *abs/1706.00764*. http://arxiv.org/abs/1706.00764

Institute-Cardiology & hospital Faisalabad-Pakistan, A. (2017). Survival analysis of heart failure patients: A case study.

Instituto-Nacional-Cáncer. (2019). *Cardiopatía coronaria* [Fecha de acceso: 27 abril de 2022]. https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/cardiopatia-coronaria

J., C.-J. C., E., C.-P. C., A., R.-G. S., Lucia, G., Liliana, M.-P., & R., R.-C. H. (2017). Factores de riesgo para enfermedad cardiovascular en adultos mexicanos. *Rev Med MD.*, *9*(2). https://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=77487

Jain, D., & Singh, V. (2018). Feature selection and classification systems for chronic disease prediction: A review. *Egyptian Informatics Journal*, *19*(3), 179-189. https://doi.org/https://doi.org/10.1016/j.eij.2018.03.002

Jaramillo, J. H., Bhadury, J., & Batta, R. (2002). On the use of genetic algorithms to solve location problems [Location Analysis]. *Computers Operations Research*, *29*(6), 761-779. https://doi.org/https://doi.org/10.1016/S0305-0548(01)00021-1

Julianna, D., & IBM. (2021). *Supervised vs. Unsupervised Learning: What's the Difference?* [Fecha de acceso: 29 abril de 2022]. https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning

Kabir, H. E., & Zaman, M. S. U. (2020). Developing a Hyperparameter Tuning Based Machine Learning Approach of Heart Disease Prediction. *7*, 631-647. https://doi.org/10.33736/jaspe.2639.2020

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, *4*, 1942-1948 vol.4. https://doi.org/10.1109/ICNN.1995.488968

Kim, Y., & Chung, M. (2019). An Approach to Hyperparameter Optimization for the Objective Function in Machine Learning. *Electronics*, *8*(11). https://doi.org/10.3390/electronics8111267

Kotthoff, L., Thornton, C., Hoos, H. H., Hutter, F., & Leyton-Brown, K. (2017). Auto-WEKA 2.0: Automatic Model Selection and Hyperparameter Optimization in WEKA. *J. Mach. Learn. Res.*, *18*(1), 826-830.

Li, J. P., Haq, A. U., Din, S. U., Khan, J., Khan, A., & Saboor, A. (2020). Heart Disease Identification Method Using Machine Learning Classification in E-Healthcare. *IEEE Access*, *8*, 107562-107582. https://doi.org/10.1109/ACCESS.2020.3001149

Luong, P., Gupta, S., Nguyen, D., Rana, S., & Venkatesh, S. (2019). Bayesian Optimization with Discrete Variables. En J. Liu & J. Bailey (Eds.), *AI 2019: Advances in Artificial Intelligence* (pp. 473-484). Springer International Publishing.

MedlinePlus. (2022a). *Accidente cerebrovascular* [Fecha de acceso: 27 abril de 2022]. https://medlineplus.gov/spanish/ency/article/000726.htm

MedlinePlus. (2022b). *Insuficiencia cardíaca* [Fecha de acceso: 27 abril de 2022]. https://medlineplus.gov/spanish/ency/article/000158.htm

Mockus, J., Tiesis, V., & Zilinskas, A. (1978). The Application of Bayesian Methods for Seeking the Extremum. *Towards Global Optimization*, *2*(117-129), 2.

Morales López, C. E., & Labrín, B. C. (2010). *Búsqueda de parámetros por optimización de enjambre de partículas para un solver de problemas de satisfacción de restricciones.*

National-Heart, L., & Blood-Institute. (2022). Framingham Heart Study-Cohort (FHS-Cohort). https://biolincc.nhlbi.nih.gov/studies/framcohort/

Nayyar, A., Le, D.-N., & Nguyen, N. G. (2018). *Advances in Swarm Intelligence for Optimizing Problems in Computer Science (1st ed.).* Chapman; Hall/CRC.

O'Donnela, C. J., & Elosuab, R. (2008). Factores de riesgo cardiovascular. Perspectivas derivadas del Framingham Heart Study. *Revista Española de Cardiología*, *61*(3). https://doi.org/10.1157/13116658

Organización-Mundial-Salud. (2021). *Hipertensión* [Fecha de acceso: 27 abril de 2022]. https://www.who.int/es/news-room/fact-sheets/detail/hypertension

Pannakkong, W., Thiwa-Anont, K., Singthong, K., Parthanadee, P., Buddhakulsomsiri, J., & Chang, K.-H. (2022). Hyperparameter Tuning of Machine Learning Algorithms Using Response Surface Methodology: A Case Study of ANN, SVM, and DBN. *Mathematical Problems in Engineering*, *2022*, 1-17. https://doi.org/10.1155/2022/8513719

Patel, P., Ordunez, P., DiPette, D., Escobar, M. C., Hassell, T., Wyss, F., Hennis, A., Asma, S., Angell, S., Treatment, S. H., & Network, P. (2016). Improved Blood Pressure Control to Reduce Cardiovascular Disease Morbidity and Mortality: The Standardized Hypertension Treatment and Prevention Project. *Journal of clinical hypertension (Greenwich, Conn.)*, *18*(12), 1284-1294. https://doi.org/https://doi.org/10.1111/jch.12861

Prabu, S., Thiyaneswaran, B., Sujatha, M., Nalini, C., & Rajkumar, S. (2022). Grid Search for Predicting Coronary Heart Disease by Tuning Hyper-Parameters. *Computer Systems Science and Engineering*, *43*(2), 737-749. https://doi.org/https://doi.org/10.32604/csse.2022.022739

Probst, P., Boulesteix, A.-L., & Bischl, B. (2018). Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *J. Mach. Learn. Res.*, *20*, 53:1-53:32. https://api.semanticscholar.org/CorpusID:88515435

Purushottam, Saxena, K., & Sharma, R. (2016). Efficient Heart Disease Prediction System [International Conference on Computational Modelling and Security (CMS 2016)]. *Procedia Computer Science*, *85*, 962-969. https://doi.org/https://doi.org/10.1016/j.procs.2016.05.288

Reddy, K. V. V., Elamvazuthi, I., Aziz, A. A., Paramasivam, S., Chua, H. N., & Pranavanand, S. (2021). Heart Disease Risk Prediction Using Machine Learning Classifiers with Attribute Evaluators. *Applied Sciences*, *11*(18). https://www.mdpi.com/2076-3417/11/18/8352

Rohit, B., Aditya, K., Mohammad, S., Gaurav, D., Sagar, P., & Parneet, S. (2021). Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning. *Computational Intelligence and Neuroscience*, (8387680), 11. https://doi.org/https://doi.org/10.1155/2021/8387680

Shah, D., Patel, S., & Bharti, S. K. (2020). Heart Disease Prediction using Machine Learning Techniques. *Sn. Comput. Sci.*, *1*. https://doi.org/https://doi.org/10.1007/s42979-020-00365-y

Texas-Heart-Institute. (2022). *Heart Disease Risk Factors* [Fecha de acceso: 30 abril de 2022]. https://www.texasheart.org/heart-health/heart-information-center/topics/heart-disease-risk-factors/

UCI. (2019). Machine Learning Repository-Cleveland Heart Disease Dataset. https://archive.ics.uci.edu/ml/datasets/Heart+Disease

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC Med Inform Decis Mak*, *19*. https://doi.org/https://doi.org/10.1186/s12911-019-1004-8

Valarmathi, R., & Sheela, T. (2021). Heart disease prediction using hyper parameter optimization (HPO) tuning. *Biomedical Signal Processing and Control*, *70*, 103033. https://doi.org/https://doi.org/10.1016/j.bspc.2021.103033

Weng, S. F., Reps, J., Kai, J., Garibaldi, J. M., & Qureshi, N. (2017). Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PLOS ONE*, *12*(4), 1-14. https://doi.org/10.1371/journal.pone.0174944

Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., & Deng, S.-H. (2019). Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimizationb. *Journal of Electronic Science and Technology*, *17*(1), 26-40. https://doi.org/https://doi.org/10.11989/JEST.1674-862X.80904120

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295-316. https://doi.org/https://doi.org/10.1016/j.neucom.2020.07.061

Zheng, A., Shelby, N., & Volckhausen, E. (2019). Evaluating Machine Learning Models. *Machine Learning in the AWS Cloud*. https://api.semanticscholar.org/CorpusID:51991287

TECNOLÓGICO
NACIONAL DE MÉXICO

cenidet®
Centro Nacional de Investigación
y Desarrollo Tecnológico