Pontificia Universidad
JAVERIANA
Cali

CLASIFICACIÓN DEL CÁNCER DE PÁNCREAS A TRAVÉS DE INFORMACIÓN CLÍNICA MEDIANTE TÉCNICAS DE APRENDIZAJE AUTOMÁTICO

Juan Pablo Ospina Cobo

Proyecto de grado

Director:
Hernán Darío Vargas Cardona

Pontificia Universidad Javeriana Cali
Facultad de Ingeniería y ciencias
Ingeniería de sistemas y computación
Santiago de Cali, mayo de 2026

RESUMEN

El cáncer de páncreas es una de las enfermedades más letales a nivel mundial debido a la dificultad de su detección temprana, causada por la ausencia de síntomas específicos, la falta de biomarcadores precisos y su compleja ubicación anatómica. Por tanto, los algoritmos de aprendizaje automático se presentan como una herramienta útil para el análisis de datos clínicos y la identificación de patrones relevantes en la información médica de los pacientes.

Por ende, el presente proyecto propone el uso de algoritmos de aprendizaje automático como herramienta para analizar información clínica obtenida de exámenes urinarios, con el fin de identificar patrones que permitan estimar la presencia de la enfermedad. Se busca así explorar un conjunto de técnicas aplicadas a un grupo de características para evaluar su relevancia, y a su vez, desarrollar una interfaz para el ingreso de información, para así, contribuir a la detección del cáncer. La metodología incluye el preprocesamiento de datos, el análisis exploratorio del conjunto de datos y la clasificación mediante aprendizaje automático. Además, se evaluará el rendimiento de los modelos utilizando exactitud, sensibilidad, precisión, especificidad, f1-score y AUC-ROC.

Palabras clave: Cáncer de páncreas, aprendizaje automático, biomarcadores, clasificación, características médicas.

ABSTRACT

Pancreatic cancer is one of the most lethal diseases worldwide due to the difficulty of its early detection, caused by the absence of specific symptoms, the lack of precise biomarkers, and its complex anatomical location. Therefore, machine learning algorithms emerge as a useful tool for the analysis of clinical data and the identification of relevant patterns in patients' medical information.

Hence, the present project proposes the use of machine learning algorithms as a tool to analyze clinical information obtained from urine tests to identify patterns that allow predicting the presence of the disease. Thus, it seeks to explore a set of techniques applied to a group of features to evaluate their relevance, while also developing an interface for data entry, thereby contributing to cancer detection. The methodology includes data preprocessing, exploratory data analysis, and classification through machine learning. In addition, the performance of the models will be evaluated using accuracy, sensitivity, precision, specificity, F1-score, and AUC-ROC.

Keywords: Pancreatic cancer, machine learning, biomarkers, classification, medical features.

Tabla de contenido

Tabla de figuras

Índice de tablas

INTRODUCCIÓN

El cáncer de páncreas es uno de los tipos de cáncer más letales a nivel mundial, principalmente debido a la baja efectividad en su detección temprana. Esto se debe a que la enfermedad no presenta síntomas específicos, carece de biomarcadores suficientemente precisos y se encuentra en una ubicación anatómica de difícil acceso. Como consecuencia, su diagnóstico suele realizarse en etapas avanzadas, lo que se traduce en una tasa de supervivencia a cinco años significativamente baja [1].

En este contexto, los algoritmos de aprendizaje automático se presentan como una herramienta útil para el análisis de datos clínicos y la identificación de patrones relevantes en la información médica de los pacientes. Su aplicación podría contribuir a mejorar la precisión en el diagnóstico del cáncer de páncreas, favoreciendo una detección más temprana y posibilitando intervenciones médicas oportunas que ayuden a reducir la tasa de mortalidad asociada.

El presente proyecto tiene como objetivo explorar técnicas de aprendizaje automático aplicadas a un conjunto de características clínicas obtenidas a partir de exámenes urinarios, con el fin de evaluar su relevancia en el diagnóstico de la enfermedad. En el documento se lleva a cabo el preprocesamiento para gestionar valores faltantes, agrupación de características, seguido de un análisis exploratorio de datos para comprender la distribución de cada característica. Posteriormente se realiza la selección de características, y un análisis comparativo de diferentes clasificadores como SVM, KNN, XGBoost, regresión logística y Random forest que permita seleccionar el que obtenga el mejor desempeño para implementarlo en una interfaz en la que un posible usuario pueda realizar predicciones a partir de sus biomarcadores.

1. DEFINICIÓN DEL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

El cáncer de páncreas (CP) constituye un grave problema de salud pública debido a su elevada mortalidad. Aunque es el décimo tipo de cáncer más frecuente, ocupa el séptimo lugar como causa principal de muerte por cáncer en el mundo [2]. En 2020 se reportaron 495,918 casos y 466,003 muertes asociadas [2], lo que evidencia que la mayoría de los diagnósticos derivan en desenlaces fatales. La tasa de supervivencia a cinco años tras el diagnóstico apenas alcanza el 11,5% [1] reflejando la baja probabilidad de sobrevida. La magnitud de la enfermedad se ilustra en que, diariamente, se diagnostican alrededor de 1,000 personas, de las cuales aproximadamente 985 fallecen [3]. La tendencia resulta aún más preocupante si se considera que la mortalidad por CP se incrementó 2.3 veces, pasando de 196,000 muertes en 1990 a 441,000 en 2017 [2]. En el contexto regional, las proyecciones para Latinoamérica y el Caribe tampoco son alentadoras, ya que se estima un aumento del 101% en la tasa de mortalidad para 2040 [2]. En conclusión, aunque el cáncer de páncreas no se encuentra entre los más comunes, su elevada letalidad lo convierte en una de las neoplasias más devastadoras, sin perspectivas de mejora en el futuro cercano.

Ahora, esta alta mortalidad no es coincidencia, y es que la razón de esto es la poca efectividad en su detección. En primer lugar, una de las causas por lo que es sumamente difícil detectarlo es que en sus etapas iniciales no presenta síntomas muy específicos, es decir que rara vez presenta síntomas distintivos [4]. Además, los síntomas pueden aparecer solapados por los de otras patologías [5] del mismo modo, aquellos que son más específicos de la enfermedad suelen aparecer una vez está muy avanzada y se ha presentado metástasis, lo cual la hace inoperable [6]. En segundo lugar, hacen falta biomarcadores de diagnóstico lo suficientemente precisos que contribuyan a la detección del CP en etapas tempranas [4], en otras palabras, no se han descubierto sustancias presentes en el cuerpo que puedan ser un indicador de la presencia de la enfermedad en una persona. En tercer lugar, debido a la ubicación del páncreas en el cuerpo humano no es posible realizar pruebas de detección rutinarias en el consultorio, además de que está en un entorno altamente vascularizado, razón por la que es más fácil una rápida metástasis [4]. Finalmente, la alta mortalidad que presenta la enfermedad se debe a la complejidad para detectarla por ciertas condiciones que presenta la misma, pero, al igual que sucede con una gran variedad de enfermedades, la mejor estrategia para mejorar la mortalidad es la detección precoz, por lo que todos los avances deben estar orientados a ese propósito.

La detección precoz constituye la estrategia más efectiva para reducir la mortalidad, y en este contexto el aprendizaje automático ofrece soluciones innovadoras tanto para la identificación temprana del cáncer de páncreas como para la detección de pacientes con alto riesgo. En primer lugar, los modelos de inteligencia artificial permiten analizar grandes volúmenes de datos de manera rápida y precisa, logrando identificar patrones asociados con el desarrollo de la enfermedad. Esto posibilita discriminar a los pacientes que podrían beneficiarse de pruebas de detección específicas, mejorando así los resultados clínicos [4]. En segundo lugar, los modelos de ML tienen la capacidad de integrar y procesar diferentes fuentes de información,

para reconocer patrones sutiles que podrían pasar desapercibidos para un observador humano. En síntesis, el aprendizaje automático ofrece ventajas significativas frente a las limitaciones humanas, al proporcionar herramientas con gran potencial para la predicción y la detección temprana del cáncer de páncreas.

1.2. OBJETIVOS

1.2.1. Objetivo general

Desarrollar un modelo de aprendizaje automático supervisado que a partir del análisis de información clínica contribuya a la detección de cáncer de páncreas.

1.2.2. Objetivos específicos

- Realizar un análisis exploratorio de bases de datos (EDA) que permita suprimir muestras anómalas e identificar las características más significativas de la información clínica disponible.
- Implementar algoritmos de aprendizaje automático supervisado para realizar la predicción.
- Evaluar la efectividad de los modelos predictivos empleando métricas estándar de evaluación para seleccionar el mejor método.
- Implementar una interfaz que facilite el ingreso de información y la visualización de resultados.

1.3. JUSTIFICACIÓN

El proyecto resulta de gran utilidad al proponer una herramienta de apoyo en el proceso de diagnóstico. En este sentido, el modelo tiene como objetivo identificar posibles factores de riesgo y, de esta manera, contribuir a la detección del cáncer de páncreas mediante el análisis de información clínica que pueda indicar la presencia de la enfermedad en el paciente. Asimismo, permite explorar diferentes características clínicas y analizar su relevancia a partir de diversos algoritmos de aprendizaje automático, lo que facilita determinar si dichas variables pueden considerarse factores importantes en el diagnóstico temprano. En síntesis, su implementación podría favorecer una identificación oportuna de la enfermedad, permitiendo una intervención médica temprana con el potencial de reducir la tasa de mortalidad [5] [6].

Por otra parte, el proyecto es viable gracias a la disponibilidad actual de tecnologías de aprendizaje automático y al acceso a conjuntos de datos médicos. En este caso, se utilizan datos obtenidos a partir de exámenes de orina [7], los cuales permiten entrenar y evaluar los modelos de manera efectiva. Además, existen diversas librerías en Python, como Scikit-learn y Keras, que facilitan la implementación de algoritmos sin necesidad de desarrollarlos desde cero. A esto se suma el acceso a equipos de cómputo con suficiente capacidad de procesamiento, lo que posibilita realizar el entrenamiento y la evaluación de los modelos de manera eficiente y a bajo costo. En conjunto, estos factores evidencian que el desarrollo del proyecto no solo es viable, sino también sostenible desde el punto de vista técnico y económico.

Finalmente, el desarrollo de este proyecto tiene el potencial de generar un impacto positivo tanto en los pacientes como en los profesionales de la salud. Por un lado, la alta letalidad del cáncer de páncreas se debe, en gran medida, a la dificultad para detectarlo en etapas tempranas, ya que la mayoría de los diagnósticos se realizan en fases avanzadas de la enfermedad. En este contexto, la implementación de un modelo de aprendizaje automático que contribuya a la predicción del cáncer a partir de ciertos factores de riesgo podría aumentar las probabilidades de detección temprana, lo cual es clave para reducir la mortalidad asociada. De hecho, estudios previos han mostrado que aproximadamente el 80% de los pacientes diagnosticados en etapas iniciales sobreviven al menos cinco años [6], lo que resalta la importancia de la detección oportuna.

Por otro lado, el uso de herramientas automatizadas para el análisis de datos permite procesar grandes volúmenes de información clínica de manera eficiente, reduciendo la carga de trabajo manual para los profesionales de la salud. De esta manera, los médicos pueden concentrar sus esfuerzos en aquellos pacientes que presentan características identificadas por el sistema como posibles indicios de la enfermedad, optimizando así la toma de decisiones clínicas.

2. MARCO DE REFERENCIA

2.1. MARCO TEÓRICO

2.1.1. Cáncer de páncreas.

Hoff [8] describe que "Los cánceres de páncreas se dividen en neoplasias del páncreas endocrino y tumores del páncreas exocrino. Estos tumores se originan en la cabeza, el cuerpo o la cola del páncreas y se caracterizan por reacciones estomacales desmoplásicas infiltrantes". Además, ahonda un poco más en la historia y física de la enfermedad y explica que "la presentación clásica para un paciente con la enfermedad es dolor abdominal y pérdida de peso, el dolor se localiza en la espalda". Por otra parte, expone que algunos elementos importantes en la historia clínica son el color claro de las heces, la aparición de diabetes en el año anterior, ictericia (piel amarillenta), depresión, y prurito (sensación de picazón).

- **PDAC (adenocarcinoma ductal pancreático):** es una neoplasia altamente agresiva. El PDAC es el tumor pancreático más común, representa alrededor del 90% de todas las neoplasias malignas pancreáticas. El PDAC se caracteriza morfológicamente por un proceso fibrótico intenso, que se conoce como reacción desmoplásica. Por otro lado, el PDAC es más común en hombres, y asimismo es más común en afroamericanos que en caucásicos. Al mismo tiempo, la edad avanzada, el tabaquismo, la inactividad física, la obesidad, la diabetes, la pancreatitis aguda, la pancreatitis crónica, y trastornos hereditarios se relacionan con un mayor riesgo de cáncer [9].
- **Enfermedad hepatobiliar:** el sistema hepatobiliar son aquellos órganos y conductos que elaboran y almacenan la bilis (líquido que produce el hígado y ayuda a digerir la grasa) y la liberan hacia el intestino delgado, está formado por el hígado, junto con las vías biliares y el páncreas. Debido a la proximidad de sus estructuras, alteraciones que ocurran en este sistema pueden repercutir en funciones de órganos vecinos [10]**.** Las enfermedades hepatobiliares son aquellas que afectan el correcto funcionamiento de este sistema, entre las cuales se encuentran algunas como: la pancreatitis, gastritis y la colecistitis.
    - **Pancreatitis:** la pancreatitis es una inflación en el páncreas, la cual puede ser aguda o crónica, sus causas pueden ser el alcoholismo, los cálculos biliares, traumatismos, ciertos medicamentos, infecciones, enfermedades hereditarias, y algunas otras desconocidas [11]. El páncreas produce enzimas para la digestión, si este se daña o se bloquea el conducto estas podrían digerir el tejido pancreático, lo cual provo inflamación y da lugar a la pancreatitis aguda [11]. Si el daño continúa, la enfermedad podría evolucionar lentamente hasta destruir gran parte del páncreas, lo cual corresponde a la pancreatitis crónica. Ambas pueden ser mortales, la pancreatitis aguda causa la muerte en el 5% de los casos, usualmente porque se dan complicaciones como destrucción extensa del tejido, hemorragias o infecciones, mientras que la pancreatitis crónica puede ser fatal en el 50% de los casos [11].

2.1.2. Machine Learning.

El machine learning (ML) o aprendizaje automático, es una rama de la inteligencia artificial que permite procesar grandes volúmenes de datos mediante algoritmos que ejecutan datos de entrada, mediante este proceso es posible predecir valores de salida en un rango específico identificando simultáneamente tendencias y patrones variables [12]. Esta tecnología permite analizar, detectar y clasificar grandes cantidades de datos muy rápidamente, más rápido de lo que lo haría una persona por si sola. Existen dos clasificaciones principales del aprendizaje automático.

- **Aprendizaje supervisado:** es una forma de ML en la que es necesario no solo ingresar los datos de entrada sino también los datos de salida en el conjunto de datos usado para entrenar el modelo. Es decir, se deben conocer los datos de salida que se obtendrían con los datos de entrada que se poseen, a partir de los patrones detectados el modelo realizará predicciones sobre datos no conocidos o sobre aquellos de los que no se poseen su salida. Algunos algoritmos de aprendizaje supervisado son:
    - **Máquina de vectores de soporte (SVM):** uno de los métodos más usados en problemas de clasificación. Tiene como objetivo maximizar el espacio entre las clases. Además, utiliza funciones kernel para manejar relaciones no lineales complejas entre las características [4] [13].
    - **Random forest:** este algoritmo es descrito en [1] como un método de aprendizaje por conjuntos que analiza datos de tiempo hasta un evento. Por otro lado, "utiliza técnicas como bootstrapping y división aleatoria de nodos para crear un grupo de árboles de decisión, y los resultados se promedian entre todos los árboles".
    - **Regresión logística (LR):** modelo estadístico común basado en la probabilidad. Sarker describe este algoritmo en [14] como un modelo que "utiliza una función logística para estimar las probabilidades". Este modelo funciona mejor cuando las clases se pueden separar linealmente.
    - **K-vecinos más cercanos:** en [14] se describe este algoritmo como un algoritmo de aprendizaje perezoso. Este algoritmo consiste en "almacenar todas las instancias correspondientes a los datos de entrenamiento en un espacio n-dimensional" [14]. La clasificación la hace a partir de calcular un voto de mayoría simple de los k-vecinos más cercanos de cada punto. El problema principal es elegir el número óptimo de vecinos.
    - **XGBoost (Impulso de gradiente extremo):** así como los bosques aleatorios, es un algoritmo por conjuntos. "Este utiliza el gradiente para minimizar la función de pérdida y utilizan el descenso del gradiente para optimizar los pesos" según Sarker en [14]. XGBoost calcula tanto gradientes de segundo orden de la función de pérdida para minimizar la misma pérdida, como la regularización avanzada para reducir el sobreajuste y mejorar la generalización y el rendimiento [14].

2.1.3. Biomarcadores.

Un biomarcador es una característica de un cuerpo biológico que puede medirse y evaluarse para determinar la salud de un ser vivo. Este puede usarse para ayudar a determinar el

progreso de una enfermedad, el éxito de un tratamiento y predecir la evolución de una enfermedad en uno o más individuos [15].

- **Creatinina:** según se expone en [16], "la creatinina es un subproducto del fosfato de creatina en el músculo, que suministra energía al tejido muscular". Según explican Babasheb et. al, es muy importante medir clínicamente los niveles de creatinina ya que son "indicadores de la funcionalidad renal, muscular y tiroidea". Este biomarcador usualmente se examina mediante análisis de sangre o de orina y tiene una alta relevancia como biomarcador para diversas enfermedades.
- **$LYVE_1$: "**es el receptor de hialuronano endotelial de los vasos linfáticos" de acuerdo con [17]. Este biomarcador está involucrado en la regulación del crecimiento celular, por lo que, según Li, "un crecimiento celular anormal, como el desarrollo del cáncer, conduce a una mayor concentración de $LYVE_1$ en la orina".
- **REG1B: "**es una proteína implicada en la regeneración celular, cuya regulación puede variar según el estado de salud o enfermedad" [18]. En [17] se describe REG1B como una proteína que "puede estar asociada con la regeneración de las células de los islotes del páncreas y puede estar involucrada en la litogénesis".
- **TFF_1:** el TFF_1 (Factor Trefoil 1) pertenece a una familia de péptidos secretores gastrointestinales que interactúan con las mucinas y se expresan en niveles aumentados durante la reconstitución y la reparación de la lesión de la mucosa [7]**.** De acuerdo con Sabernardi et al. este péptido protege a las células epiteliales de la muerte apoptótica y aumentan su motilidad, pero también tiene funciones similares en las células cancerosas, por tal motivo están involucradas en el desarrollo de varios tipos de cáncer.
- **REG1A:** en [19] se describe como un miembro de la familia de genes regenerativos, el cual participa en la proliferación celular, diferenciación y regeneración tisular. Desde que se descubrió se ha demostrado que participa en la lesión pancreática, inflamación y metaplasia acinar a ductal. Niveles séricos elevados de REG1A están asociados con pancreatitis con datos limitados de pacientes, lo que indica que este podría reflejar el daño del páncreas [19].
- **CA19-9: "**el antígeno carbohidrato 19-9 es un biomarcador e indicador de glicosilación aberrante en el cáncer de páncreas" [20]. En la investigación [20] se plantea que este antígeno funciona como biomarcador, predictor y promotor en el cáncer de páncreas. En primer lugar, tiene potencial de detección cuando se combina con síntomas y/o factores de riesgo. En segundo lugar, como predictor puede usarse para evaluar el estadio, el pronóstico, la resecabilidad, recurrencia y eficacia terapéutica. Por último, como promotor, se puede usar para evaluar la biología del cáncer. Este podría acelerar la progresión del cáncer a través de la glicosilación de proteínas, la unión a la E-selectina, el fortalecimiento de la angiogénesis y la mediación de la respuesta inmunológica.

2.1.4. Análisis exploratorio de datos.

El análisis exploratorio de los datos (EDA) es un proceso que permite identificar patrones, correlaciones y resumir características principales, usualmente con métodos de visualización de datos, que ayuden a determinar cuáles variables son relevantes. Además, ayuda a

identificar irregularidades y anomalías, como valores atípicos, que podrían deberse a una mala calidad de los datos [21]. En resumen, permite comprender mejor las variables del conjunto de datos y las relaciones entre ellas y prepararlos para el uso.

- **Rango intercuartílico (IQR):** la mediana muestral divide el conjunto de datos en dos partes más o menos iguales, de forma que la mitad de los elementos son menores y la otra mitad mayores que ella. En lugar de identificar solo el centro del conjunto de datos, para entender mejor la variabilidad de ellos se sugiere proporcionar un resumen de cinco números: el mínimo, máximo, mediana muestral, percentil 25% y percentil 75%. El percentil 25% (Q1) se llama cuartil inferior y el percentil 75% (Q3) se llama cuartil superior. Con la mediana (Q2), estos cuartiles dividen el conjunto de datos en cuatro partes aproximadamente iguales, cada una cuenta con cerca de una cuarta parte de los elementos. La diferencia entre el cuartil superior (Q3) y el inferior (Q1) es lo que se conoce como rango intercuartílico, el cual representa el rango de la mitad central de los datos y sirve como medida de la variabilidad [22].

2.1.5. Métricas de rendimiento.

Para las tareas de clasificación binaria, los resultados se resumen en una matriz, conocida como matriz de confusión, que divide los resultados de acuerdo con la clasificación en las siguientes categorías: TP (True Positives), muestras positivas clasificadas como positivas; TN (True Negatives), muestras negativas clasificadas como negativas; FP (False Positives), muestras negativas clasificadas como positivas; FN (False Negatives), muestras positivas clasificadas como negativas.

Figura 1 Matriz de confusión

- **Exactitud:** fracción de las muestras clasificadas correctamente [23]. Para calcularla se aplica la siguiente fórmula:

$$\frac{TP + TN}{TP + FP + TN + FN}$$

- **Sensibilidad:** proporción de casos positivos que han sido correctamente identificados por el modelo [23]. Sirve para cuantificar la sensibilidad del modelo para detectar muestras positivas entre un conjunto de negativas. Se calcula así:

$$\frac{TP}{TP + FN}$$

- **Especificidad:** proporción de muestras negativas que fueron realmente clasificadas como negativas [23]. Indica que es efectiva para identificar con precisión las muestras negativas. Se calcula de la siguiente forma:

$$\frac{TN}{TN + FP}$$

- **Precisión:** proporción de muestras clasificadas como positivas que realmente son positivas [23]. Sirve para identificar cuántas de las clasificaciones positivas son correctas, lo cual indica que tan confiable son las clasificaciones positivas. Se mide de 0 a 1 y se calcula así:

$$\frac{TP}{TP + FP}$$

- **F1-score:** es la media armónica entre la precisión y la sensibilidad [23]. Se mide de 0 a 1, un F1-score igual a 1 indica que el modelo es perfecto, pues en ese caso tanto precisión como sensibilidad es igual a 1. Un F1-score menor a 0.5 indica que el modelo tiene problemas, pues es indicio de que hay muchos falsos positivos y negativos.

$$2\frac{Precisión \times Sensibilidad}{Precisión + sensibilidad}$$

- **AUC-ROC (área bajo la curva ROC):** esta métrica sirve para conocer la probabilidad de que una muestra positiva tenga una puntuación de clasificación más alta que una muestra negativa. Un AUC-ROC de 1 es una clasificación perfecta y un AUC-ROC de 0.5 es que no tiene capacidad de discriminación [23], es decir, que no distingue entre las clases siendo así, un modelo aleatorio. En resumen, el AUC-ROC sirve para medir qué tan bien un modelo de clasificación binaria distingue entre las clases.

2.1.6. Validación cruzada.

La validación cruzada es una estrategia ampliamente utilizada en el aprendizaje automático. Resulta de gran utilidad para evitar sesgos y el sobreajuste, y es fundamental en la evaluación del rendimiento de los modelos. Esta técnica consiste en entrenar el modelo con todos los pliegues excepto uno, que se reserva para la validación. El proceso se repite varias veces, de modo que cada pliegue actúe como conjunto de prueba una sola vez [24]. La evaluación repetida en diferentes subconjuntos de datos permite comprobar la capacidad del modelo para generalizar ante entradas desconocidas. Además, la validación cruzada ayuda a mitigar la variabilidad que puede surgir al depender de una única partición de entrenamiento y prueba [24].

- **Leave One Out (LOOCV):** la técnica de Leave One Out o Dejar Uno Afuera por su traducción al español, consiste en omitir un único registro o fila en cada entrenamiento o proceso de aprendizaje. Teniendo n como el número de registros en el conjunto de datos, el entrenamiento se realiza con n-1 registros y la validación con un solo registro. Cada registro es usado para validación una única vez. Este proceso se repite n veces, es decir tantas veces como registros tenga el conjunto de datos [25].
- **KFOLD:** es un método de la familia de técnicas de validación cruzada, consiste en particionar los datos en varios subconjuntos y utilizar sistemáticamente diferentes subconjuntos para entrenamiento y validación, lo cual ofrece un marco robusto para la evaluación de modelos [26]. Es decir, se divide el conjunto de datos en n subconjuntos, y realiza n iteraciones donde, en cada iteración utiliza uno de los subconjuntos como conjunto de prueba y los otros n-1 subconjuntos como entrenamiento, cada subconjunto se usa como prueba una única vez. Este enfoque es de gran importancia en los campos biomédicos, donde el tamaño de la muestra puede ser limitado [26].
- **Hold-Out:** en este método los datos se dividen en dos partes que no se superponen y se utilizan para entrenamiento y prueba. La parte que se utiliza para la prueba se denomina la parte hold-out. Recibe ese nombre porque se reserva esa parte para la prueba y se entrena al modelo con el resto de los datos. El método hold-out puede utilizar diferentes porcentajes de datos reservados para las pruebas [27].

2.1.7. Normalización.

Es el proceso de transformar datos a un rango específico, como entre 0 y 1. Esta es necesaria cuando existen grandes diferencias en los rangos de diferentes características. Se ha utilizado para abordar problemas de valores atípicos y características dominantes, de forma que reduce los efectos que estos pueden causar al lograr uniformidad numérica en los datos. Al normalizar se garantiza que cada característica tenga una misma contrición cuando se introduce en el clasificador. Existen métodos de escalado que re-escalan los datos a un rango predeterminado de forma que conservan la distribución de datos inicial. Por otro lado, existen otros métodos de transformación que escalan los datos a un rango común pero no conservan la distribución de datos inicial. Estos métodos mencionados desplazan ciertos valores hacia el centro de la distribución mientras se escalan los datos a un rango fijo [28] [29].

- **MaxAbsScaler:** consiste en escalar las características de modo que el valor absoluto máximo de cada característica se escale al tamaño de la unidad. Centrar datos dispersos destruiría la estructura de dispersión, así que rara vez es sensato hacerlo, sin embargo, puede tener sentido si las características se encuentran en escalas diferentes. “MaxAbsScaler fue diseñado específicamente para escalar datos dispersos, y esta es la forma recomendada de hacerlo”. Funciona escalando los datos de entrenamiento para que queden dentro del rango [-1, 1], dividiendo por el valor máximo absoluto de cada característica [30].
- **RobustScaler:** función recomendada por Scikit learn en caso de que los datos contengan muchos datos atípicos, para ello, expresa textualmente que “Si sus datos contienen muchos valores atípicos, escalar utilizando la media y la varianza

probablemente no funcione bien. En estos casos, puede usar RobustScaler como reemplazo directo. Este método utiliza estimaciones más robustas para el centro y el rango de sus datos" [30].

2.2. ANTECEDENTES

2.2.1. Diagnóstico de Cáncer de Páncreas a partir de Imágenes de Tomografía Computarizada (TC) Utilizando Métodos de Machine Learning.

En este trabajo [13], Kashikar et al. implementaron distintos modelos para clasificar imágenes de TC, con la finalidad de mejorar la precisión el diagnóstico del cáncer, para ello implementaron y compararon el algoritmo de Máquina de vectores de soporte (SVM) y redes neuronales convolucionales (CNN). Para realizar la clasificación tomaron un dataset de 5379 imágenes. Al implementar el modelo de SVM obtuvieron un 83.07% de precisión, mientras que con el modelo de CNN obtuvieron un 90.01% de precisión y un 85% de sensibilidad. A partir de ello concluyeron que el algoritmo CNN fue el de mejor rendimiento, y que por lo tanto esta herramienta es más efectiva para la detección del cáncer que el SVM y que otros enfoques tradicionales de ML, sobre todo al trabajar con imágenes.

Este trabajo, aunque difiere del propio ya que, en aquel se extrajeron las características a partir de imágenes, podría ser un aporte muy importante al proceso de desarrollo ya que usaron un algoritmo que será usado en este proyecto, por lo que podría ser útil aprender del proceso que siguieron en la configuración de los hiperparámetros del algoritmo para lograr el mejor resultado posible, pues lo ideal sería conseguir un buen rendimiento para cada uno de los algoritmos antes de elegir el mejor de ellos.

2.2.2. Localización automática de regiones tumorales pancreáticas en secuencias completas de procedimientos de ecoendoscopia

Este trabajo presenta un método para localizar automáticamente tumores pancreáticos en videos grabados de procedimientos de ecoendoscopia (EUS). Para ello dividen el proceso en tres grandes etapas: la primera constó de preprocesar fotogramas del video mediante la reorganización de las intensidades radiales capturadas, filtraron el ruido de moteado y mejoraron el contraste; en la segunda detectaron aquellos fotogramas que tuvieran tumores mediante un clasificador, usando Maquina de Vectores de Soporte o Resnet18; en la tercera usaron un modelo YOLO para predecir la localización aproximada de los tumores en los fotogramas. El conjunto de datos que usaron contenía 66.249 fotogramas, divididas en 18 casos diagnosticados con cáncer, 5 de pancreatitis y 32 con páncreas sano. Como resultado, obtuvieron una precisión del 85,3% de acuerdo con unos cálculos llevados a cabo por ellos para medir dicha precisión [31].

Por un lado, como se mencionó anteriormente, los modelos de clasificación podrían emplearse directamente en el proyecto o bien aprovecharse parcialmente, tomando de ellos procesos o enfoques que contribuyan a mejorar el rendimiento del modelo elegido. Por otro lado, la principal diferencia con el trabajo citado radica en el proceso de extracción de características a partir de imágenes: en dicho estudio se implementaron algoritmos para preprocesar fotogramas y localizar tumores en imágenes médicas, mientras que en el presente proyecto

esto no será necesario, ya que se utilizará un conjunto de datos que incluye campos con información clínica estructurada, lo que evita la necesidad de extraer datos directamente de imágenes.

2.2.3. Un marco de aprendizaje automático para la clasificación del cáncer de hígado utilizando características clínicas.

En este documento [32] Singh et al. presentan un marco de aprendizaje automático diseñado para la clasificación del cáncer de hígado utilizando características clínicas, con el objetivo de detectar el cáncer de hígado en etapas iniciales. Los investigadores implementaron un flujo de trabajo organizado que consistió en recopilar los datos, preprocesamiento, análisis exploratorio, selección de características, entrenamiento y validación de los modelos e interpretabilidad para medir el efecto de cada característica clínica en las predicciones del modelo. En el estudio se evaluaron cinco familias de algoritmos implementados en Python: SVM, KNN y Modelos de ensamble como ADABoost, XGBoost y LightGBM, además de usar Grid Search para el ajuste de hiperparametros. En cuanto a los resultados, encontraron que el modelo LightGBM obtuvo el mejor desempeño general con una exactitud del 87.24% y una sensibilidad del 87.24%, mientras que AdaBoost fue el más eficaz en reducir los falsos negativos. Además, encontraron que los factores más influyentes para identificar el cáncer fueron los niveles de alfa-feto proteína (AFP), el índice de masa corporal (IMC), la edad, el historial de cirrosis y las puntuaciones de la función hepática. Por último, el estudio concluye que este marco no solo logra una precisión competitiva, sino que también identifica factores de riesgo interpretables que pueden ayudar a los médicos en la detección temprana y en la medicina personalizada.

Este trabajo podría resultar muy útil más allá del conjunto de modelos de aprendizaje utilizados y de la organización de sus parámetros, ya que en el documento se plantea un flujo de trabajo estructurado y fácilmente reproducible que puede servir como una base importante para el desarrollo de este proyecto. Además, el análisis exploratorio de datos realizado para comprender el conjunto de datos fue bastante completo, por lo que varias de las estrategias implementadas podrían adaptarse y aplicarse en el proceso de desarrollo propuesto. Por otra parte, el documento emplea una función para identificar los predictores más relevantes durante la etapa de selección de características, herramienta que podría ser de gran utilidad en este proyecto con el mismo propósito. Asimismo, utilizaron una técnica de validación robusta como K-Fold, la cual podría aportar significativamente al proceso de evaluación si se reproduce dentro de la metodología. En resumen, este trabajo ofrece una perspectiva adicional para la construcción del flujo de trabajo, además de un conjunto de técnicas y herramientas que, al ser implementadas, podrían resultar altamente relevantes y beneficiosas para el desarrollo y la evaluación del proyecto, contribuyendo al cumplimiento de los objetivos establecidos.

3. METODOLOGÍA

3.1. SELECCIÓN DEL CONJUNTO DE DATOS

Se seleccionó un conjunto de datos proveniente del estudio "*A combination of urinary biomarker panel and PancRISK score for earlier detection of pancreatic cancer: A case–control study*" de Debernardi et. al. [7]. La investigación se diseñó como un estudio de casos y controles en el cual utilizaron muestras de orina y plasma. Se analizaron un total de 590 muestras de orina, las cuales se desglosaron de la siguiente manera:

- **183 controles:** corresponden a individuos sin enfermedades pancreáticas, malignas o renales conocidas.
- **208 enfermedades hepatobiliares benignas:** incluyen 119 casos de pancreatitis crónica, 54 enfermedades en la vesícula biliar, 20 lesiones quísticas y 15 casos con dolor abdominal de posible origen pancreático.
- **199 pacientes con adenocarcinoma ductal pancreático (PDAC):** divididos en 102 estadios tempranos (I-II) y 97 en estadios avanzados (III-IV).

Las muestras clínicas fueron obtenidas de múltiples centros internacionales, entre los cuales se encuentran el Barts Pancreas Tissue Bank, la Universidad de Liverpool, el Centro Nacional de Investigaciones Oncológicas (CNIO) en España, la Universidad de Belgrado, el Hospital Universitario de Cambridge. Además, cada uno de los procedimientos contó con la aprobación de los comités de ética institucionales y el consentimiento por escrito de todos los pacientes participantes.

En cuanto a la recolección y el almacenamiento de las muestras, para las muestras de orina se recolectó orina de chorro medio, se mantuvo en hielo y se alícuota antes de congelarse a -80°C en un plazo de 2 horas tras la recolección. Al mismo tiempo se obtuvieron muestras de sangre, se procesaron en tubos EDTA y se centrifugaron para obtener plasma, el cual también se almacenó a -80°C.

Por lo que se refiere a los datos, el dataset seleccionado cuenta con 14 columnas, entre las cuales se incluyen un identificador y variables demográficas, como la cohorte del paciente, la institución de la cual se obtuvo la muestra, la edad y el sexo.

Además, contiene información relacionada con el diagnóstico: los registros con diagnóstico 1 corresponden a muestras sin enfermedades pancreáticas; aquellos con diagnóstico 2 describen una enfermedad hepatobiliar benigna, la cual se detalla en una columna adicional; y, por otra parte, los pacientes con diagnóstico 3 presentan adenocarcinoma ductal pancreático (PDAC), para quienes también se especifica en otra columna la etapa en la que se encontraba el cáncer al momento del diagnóstico.

Por último, el conjunto incluye seis columnas correspondientes a los biomarcadores evaluados en el estudio: plasma CA19-9, creatinina, LYVE1, REG1B, TFF1 y REG1A.

Este conjunto de datos se seleccionó, en primer lugar, por ser de acceso público, lo que facilitó su disponibilidad. A pesar de los intentos por acceder a bases de datos más grandes provenientes de instituciones extranjeras, no fue posible obtenerlas: en algunos casos, las solicitudes no fueron respondidas al momento de iniciar el proyecto, mientras que en otros se

requerían documentos adicionales que dificultaban el proceso de acceso. Como consecuencia, solo se contó con un número limitado de conjuntos de datos públicos. Entre ellos, algunos presentaban información sintética o de baja confiabilidad, por lo que fueron descartados. En este contexto, la base de datos finalmente fue seleccionada por la clara especificación de cómo se obtuvieron y procesaron los datos, así como por su mayor acercamiento con los objetivos planteados en el proyecto.

3.2. ANÁLISIS EXPLORATORIO DE LOS DATOS

En esta etapa se llevó a cabo el análisis exploratorio de los datos con el objetivo de comprender de manera práctica el comportamiento del conjunto de datos utilizado. Más allá de la definición teórica, este proceso permitió examinar cómo se distribuyen las variables, identificar posibles inconsistencias y observar relaciones relevantes entre ellas que pudieran influir en el modelado.

A lo largo del EDA se analizaron aspectos como la presencia de valores atípicos, la distribución de los biomarcadores y su variación entre las diferentes clases de diagnóstico, así como la existencia de datos faltantes y su impacto en el conjunto. Este análisis también facilitó reconocer qué variables podrían aportar más información al modelo y cuáles requerían algún tipo de tratamiento previo.

En conjunto, esta etapa sirvió como base para tomar decisiones informadas sobre la limpieza, transformación y preparación de los datos, permitiendo continuar con el proceso de manera más estructurada y con un mejor entendimiento del problema.

3.2.1. Valores nulos

Como primer paso, se evaluó qué decisión tomar respecto a los datos nulos presentes en el conjunto de datos. Por un lado, se identificaron valores nulos en columnas cuya información dependía de otra variable. Por ejemplo, en los registros con diagnóstico 2 (enfermedad hepatobiliar benigna) se incluía una columna adicional denominada “benign_sample_diagnosis”, que especificaba el diagnóstico particular del paciente. En consecuencia, los registros con un valor distinto de 2 en la variable de diagnóstico presentaban valores nulos en dicha columna, lo cual es coherente con la estructura del dataset y no necesariamente indica ausencia de información relevante.

Por otro lado, se detectaron valores nulos en columnas correspondientes a los biomarcadores, lo que indicaba que en ciertos registros no se había medido alguno de ellos. Este tipo de valores nulos representó un mayor desafío, ya que estos registros podían ser importantes para el entrenamiento de los modelos. En estos casos, fue necesario evaluar distintas estrategias, como la imputación de valores (por ejemplo, mediante la media, la mediana o modelos más avanzados) o la eliminación de registros incompletos, considerando el impacto de cada decisión en el desempeño y la generalización del modelo.

En consecuencia, resultó fundamental tratar adecuadamente todos los valores nulos del dataset, con el fin de obtener un conjunto de datos más limpio, coherente y fácil de analizar, que facilitara su uso en las etapas posteriores de modelado y evaluación.

A continuación, se describe el proceso seguido para el tratamiento de los valores nulos presentes en el conjunto de datos.

En primer lugar, se identificaron dos columnas cuya información dependía directamente de la variable de diagnóstico. Una de ellas fue “benign_sample_diagnosis”, previamente descrita. En este caso, se decidió imputar los valores nulos en función del valor registrado en la columna de diagnóstico: si el registro presentaba el valor 1, el valor nulo se reemplazaba por “No cancer”, mientras que, si presentaba el valor 3, se sustituía por “Cancer”. Esta decisión permitió mantener la coherencia semántica de los datos y evitar la pérdida de información. La segunda columna fue “Stage”, la cual indica la etapa del cáncer en el momento del diagnóstico. Esta variable presentaba valores nulos en aquellos pacientes que no padecían cáncer de páncreas. Por ello, se optó por reemplazar dichos valores por “No cancer”, con el fin de unificar los criterios y facilitar el análisis posterior. De esta manera, se logró eliminar la totalidad de los valores nulos presentes en ambas columnas, asegurando un conjunto de datos más consistente y adecuado para las etapas posteriores del análisis y modelado.

En segundo lugar, se identificaron valores nulos en dos columnas correspondientes a los biomarcadores Plasma CA19-9 y REG1A. Estas variables son de tipo numérico y, además, resultan relevantes para el entrenamiento de los modelos, lo que hizo más compleja la toma de decisiones sobre su tratamiento. En una primera instancia, se consideró eliminar los registros que contenían valores nulos; sin embargo, esta opción fue descartada debido al tamaño limitado del conjunto de datos. Reducir aún más la cantidad de registros podría afectar negativamente la capacidad de entrenamiento y la generalización de los modelos. Por esta razón, se optó por imputar los valores faltantes en lugar de eliminarlos. Para ello, se evaluaron medidas estadísticas como la media y la mediana de cada una de las columnas. La elección entre ambas dependió de la distribución de los datos y de la presencia de valores atípicos, ya que la mediana suele ser más robusta en presencia de estos. Este enfoque permitió conservar la mayor cantidad de información posible sin comprometer significativamente la calidad de los datos.

Por tanto, antes de tomar una decisión, como ya se mencionó, se analizaron los valores atípicos y las distribuciones de las variables antes de tomar una decisión.

Al analizar el boxplot de la característica Plasma CA19-9 (Figura 3), se evidencia una alta presencia de valores atípicos, lo que sugiere el uso de la mediana como estrategia de imputación para los datos faltantes. Esto se debe a que la media puede verse fuertemente influenciada por valores extremos: aunque estos sean pocos, su magnitud eleva considerablemente el promedio, haciendo que no represente adecuadamente la tendencia central de los datos. Esta observación se refuerza en la Figura 2, donde se aprecia que los valores bajos son más frecuentes, mientras que los valores altos, aunque escasos, alcanzan magnitudes significativamente elevadas. Para evaluar el impacto de estas medidas, se utilizó el rango intercuartílico con el fin de estimar la proporción de valores atípicos en tres escenarios: con datos nulos, imputando con la media y con la mediana. Los resultados muestran que, con datos nulos, el 9,32% de los valores son atípicos; al imputar con la mediana, este porcentaje aumenta a 21,36%; mientras que con la media disminuye a 5,25%. Esto indica

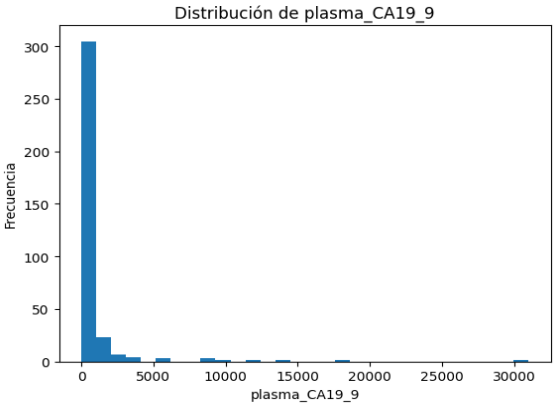

Figura 2 Distribución de Plasma CA19-9

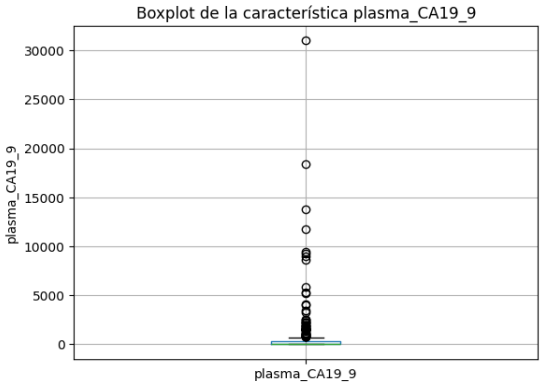

Figura 3 Boxplot de plasma CA19-9

que la media se ve afectada por los valores extremos, reduciendo artificialmente la proporción de valores atípicos, mientras que la mediana, aunque incrementa dicha proporción, conserva mejor la estructura real de los datos. En este sentido, la mediana resulta más adecuada para los modelos, ya que permite mantener patrones más representativos del comportamiento original de la variable. En conclusión, se optó por utilizar la mediana para reemplazar los valores nulos, debido a que ofrece una mejor representación de la tendencia central sin verse distorsionada por valores atípicos.

La siguiente característica con valores nulos fue el biomarcador REG1A, en el cual se observa un comportamiento similar al de la variable analizada previamente. Al examinar el diagrama de caja y bigotes (Figura 4) y su distribución (Figura 5), se evidencia una alta presencia de valores atípicos en el extremo superior. Aunque la mayor frecuencia se concentra en valores bajos, la magnitud de los valores altos puede influir considerablemente en el promedio, elevándolo y distorsionando la representación de la tendencia central. Para validar este comportamiento, se aplicó el mismo procedimiento utilizado anteriormente: se calculó la proporción de valores atípicos mediante el rango intercuartílico en tres escenarios: con los datos originales, imputando con la media y con la mediana. Los resultados obtenidos fueron los siguientes: en el conjunto original, el porcentaje de valores atípicos fue de 7,29%; al imputar con la media, se obtuvo un 7,12%; y al utilizar la mediana, la proporción aumentó a 45,42%. Estos resultados confirman que la media se ve afectada por la presencia de valores extremos, reduciendo artificialmente la proporción de valores atípicos y, por ende, alterando la estructura real de los datos. Por el contrario, aunque la mediana incrementa dicha

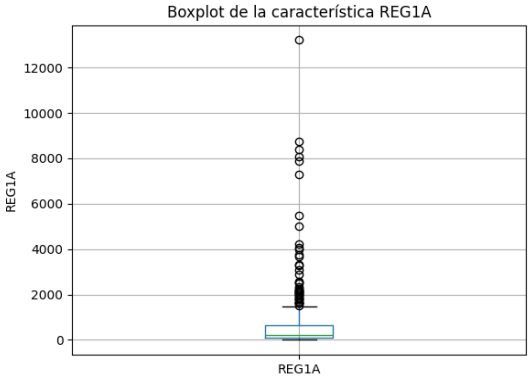

Figura 4 Boxplot de REG1A

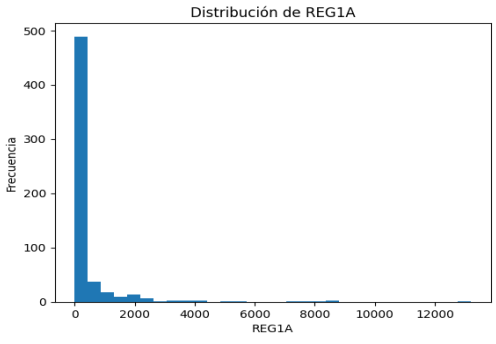

Figura 5 Distribución REG1A

proporción, preserva de mejor manera la distribución original y los patrones subyacentes. En consecuencia, se tomó la decisión de imputar los valores nulos utilizando la mediana, con el objetivo de no distorsionar la tendencia central ni comprometer la calidad de los datos para el entrenamiento de los modelos.

3.2.2. Agrupación de clases

En este proyecto, la variable objetivo (es decir, la clase que se busca predecir) corresponde al diagnóstico, representado en el conjunto de datos como "diagnosis". Esta variable cuenta con tres categorías, previamente descritas: (1) individuos sin enfermedades pancreáticas, malignas o renales conocidas; (2) pacientes con enfermedad hepatobiliar benigna; y (3) pacientes con adenocarcinoma ductal pancreático (PDAC). Cabe destacar que el objetivo principal del proyecto es implementar modelos de aprendizaje automático orientados a la detección del cáncer de páncreas. En este sentido, se planteó un enfoque de clasificación binaria, en el cual el modelo determine únicamente la presencia o ausencia de cáncer a partir de las características disponibles. Para lograrlo, se consideró la necesidad de realizar una fusión de clases, con el fin de reducir las tres categorías originales a dos. Inicialmente, se propuso agrupar las clases 2 y 3, diferenciándolas de la clase 1, de modo que se distinguiera entre pacientes sanos y pacientes con algún tipo de enfermedad. Sin embargo, tras una revisión más detallada, se identificaron diferencias significativas entre las enfermedades hepatobiliares benignas y el adenocarcinoma ductal pancreático (PDAC) en su naturaleza clínica. Esto llevó a reconsiderar dicha agrupación, ya que podría introducir ambigüedad en el proceso de clasificación y afectar negativamente el desempeño del modelo.

Concretamente, la enfermedad más similar al PDAC es la pancreatitis; sin embargo, incluso en este caso existen diferencias importantes. Por un lado, en la pancreatitis el tejido pancreático presenta cicatrices de origen alcohólico o hereditario; además, es frecuente la presencia de cálculos en el conducto pancreático y puede estar acompañada de pseudoquistes, generalmente extra pancreáticos, lo que hace que, desde un punto de vista clínico, la enfermedad se caracterice por episodios dolorosos [33]. Por otro lado, el PDAC se manifiesta como un tumor que, en la mayoría de los casos, forma una masa sólida de entre 2 y 3 cm de diámetro en la cabeza del páncreas; solo ocasionalmente presenta necrosis y, a diferencia de la pancreatitis, no se observan cálculos ni pseudoquistes [33].

Si bien existe una asociación entre la pancreatitis y el cáncer de páncreas, estudios como el de Alhobaya et al. han concluido que el riesgo de desarrollar PDAC en pacientes con pancreatitis es aproximadamente el doble en comparación con la población general durante un periodo de seguimiento de 5 a 10 años. No obstante, otras investigaciones señalan que el riesgo acumulado de PDAC en pacientes con pancreatitis crónica es del 1,8% y del 4% a los 10 y 20 años, respectivamente [34], lo que indica que, aunque existe relación, la probabilidad no es particularmente elevada.

Finalmente, con base en estas consideraciones, se tomó la decisión de realizar una fusión de clases agrupando las categorías 1 y 2. De este modo, se obtuvo una clasificación binaria en la que una clase representa a los pacientes con cáncer de páncreas (PDAC), etiquetada con el valor 1 en el conjunto de datos, y la otra agrupa a los pacientes sin cáncer, etiquetada con el

valor 0. Esta decisión permitió alinear el problema con el objetivo principal del proyecto, centrado en la detección del cáncer.

3.2.3. Elección de características

Antes que todo, aunque pueda parecer innecesario, es importante aclarar que existen variables en el conjunto de datos que, por su naturaleza, no se consideraron como candidatas para el entrenamiento de los modelos. En particular, se excluyeron las características “stage” y “benign_sample_diagnosis”, ya que dependen directamente del valor del diagnóstico. Dado que el objetivo del proyecto es predecir dicha variable, incluirlas implicaría una fuga de información, lo que sesgaría los resultados y produciría un rendimiento artificialmente alto.

Por otra parte, también se decidió excluir la variable “sexo”. Aunque algunos estudios indican que el cáncer de páncreas puede ser más frecuente en hombres que en mujeres [9], la diferencia en la tasa de incidencia global no es especialmente marcada; por ejemplo, en 2018 se reportaron aproximadamente 5,5 casos por cada 100.000 hombres frente a 4,0 por cada 100.000 mujeres [2]. Además, estas tasas pueden variar con el tiempo y entre regiones [2], lo que convierte al sexo en una variable potencialmente inestable y con un poder discriminativo limitado en este contexto.

En consecuencia, se optó por no incluir estas variables en el entrenamiento, con el fin de evitar sesgos, reducir el riesgo de sobreajuste y garantizar que el modelo aprenda a partir de características más relevantes y generalizables.

Por otra parte, una de las variables que sí se decidió incluir, a partir de la revisión de diversos estudios, fue la edad. Esto se debe a que existen diferencias significativas en la incidencia del cáncer de páncreas según este factor, ya que se trata principalmente de una enfermedad que afecta a personas de mayor edad. En particular, en Estados Unidos, el 89,4% de los nuevos casos y el 92,6% de las muertes por cáncer de páncreas ocurren en pacientes mayores de 55 años [2]. Si bien, el diagnóstico de cáncer de páncreas es extremadamente raro antes de los 30 años, se ha observado un aumento en la carga de la enfermedad debido al envejecimiento

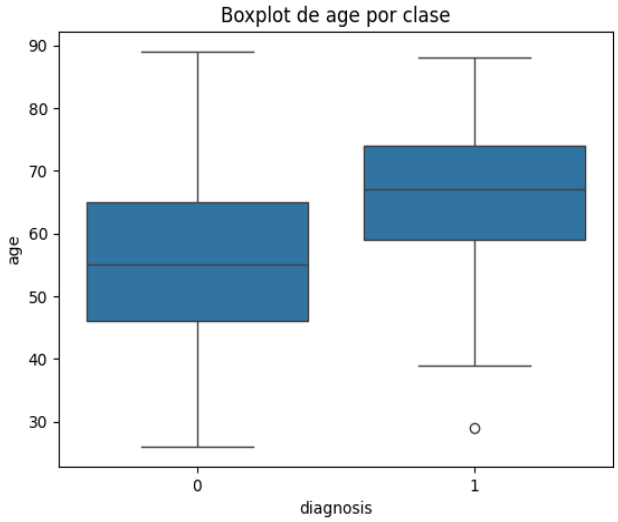

Figura 6 Boxplot edad/diagnóstico

de la población [2] [35]. En definitiva, los diagnósticos se presentan con mayor frecuencia en personas entre los 65 y 74 años, con una edad mediana de 70 años [2].

Este comportamiento es consistente con lo observado en el conjunto de datos analizado. En el boxplot mostrado en la Figura 6, la edad mediana de los pacientes diagnosticados con cáncer es cercana a los 70 años, específicamente de 67 años. Asimismo, al comparar la distribución de la edad entre pacientes con y sin cáncer, se aprecia una diferencia en sus tendencias: aunque existe cierto solapamiento entre ambos grupos, los pacientes diagnosticados con cáncer (valor 1) tienden a concentrarse en edades más altas en comparación con aquellos sin cáncer (valor 0).

Esta diferencia sugiere que la edad es una variable con capacidad discriminativa, lo que la convierte en un atributo relevante para los modelos de clasificación. En consecuencia, su inclusión en el entrenamiento contribuye a mejorar la capacidad del modelo para identificar patrones asociados a la presencia de la enfermedad.

A continuación, se aborda la selección de los biomarcadores presentes en el conjunto de datos. El primero de ellos es LYVE1, el cual, como se mencionó previamente, corresponde al receptor de hialuronano endotelial de los vasos linfáticos. Su inclusión resultó bastante clara por varias razones. En primer lugar, diversos estudios señalan que una alta concentración de LYVE1 en la orina puede estar asociada con un crecimiento celular anormal, lo que podría relacionarse con el desarrollo de enfermedades como el cáncer [17]. Desde una perspectiva conceptual, esto sugiere que este biomarcador podría tener capacidad discriminativa, ya que valores elevados podrían indicar una mayor probabilidad de presencia de la enfermedad.

Adicionalmente, se analizó su comportamiento dentro del conjunto de datos. Para ello, se construyó un diagrama de caja que permitiera visualizar su distribución en pacientes con y sin cáncer. A partir de la observación del diagrama de cajas (Figura 7) se sacaron diversas conclusiones importantes que aportaron en la decisión de incluirlo en el entrenamiento.

En primer lugar, se observa una diferencia clara en la mediana entre ambas clases: los

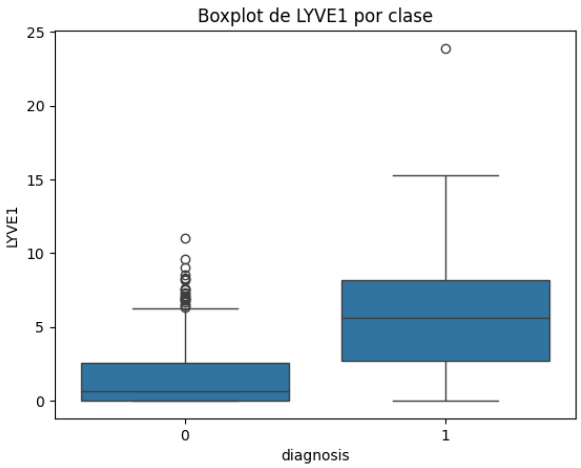

Figura 7 Boxplot LYVE1/diagnosis

pacientes con cáncer (valor 1) presentan valores centrales considerablemente más altos que

los pacientes sin cáncer (valor 0). Esto sugiere que LYVE1 tiene una buena capacidad para diferenciar entre las clases. Por otra parte, el rango intercuartílico (la caja) de la clase positiva está desplazado hacia valores más altos y es más amplio, lo que indica además de mayor concentración de valores elevados, mayor variabilidad en pacientes con cáncer. Mientras que la clase negativa presenta una distribución más concentrada en valores bajos. También se aprecia que existe cierto solapamiento entre las cajas, aunque no es predominante. Esto implica que, si bien LYVE1 no separa perfectamente las clases por sí solo, sí aporta información relevante que puede ser aprovechada en conjunto con otras variables. En resumen, estas observaciones indican que LYVE1 es una característica que podría ser muy importante, ya que muestra diferencias claras en tendencia central y dispersión entre las clases, lo que la convierte en una variable útil para el proceso de clasificación.

Por último, tanto la evidencia teórica como el análisis exploratorio respaldan la inclusión de LYVE1 como una variable relevante. Su comportamiento sugiere que puede aportar información valiosa al modelo, facilitando la diferenciación entre pacientes con y sin cáncer durante el proceso de clasificación.

El siguiente biomarcador que se incluyó fue el TFF1 o Factor Trefoil 1, que, como ya se mencionó pertenece a una familia de péptidos secretores gastrointestinales que interactúan con las mucinas. La familia de los TFF desempeña un papel importante en el mantenimiento y la protección de la integridad de la superficie epitelial, pues son secretados en respuesta a lesiones, facilitan la migración celular hacia ella, siendo cruciales en la restitución epitelial [36].

Al revisar investigaciones previas sobre este biomarcador, se evidenció su potencial relevancia para diferenciar entre pacientes con y sin la enfermedad. Diversos estudios señalan que se trata de un péptido involucrado en procesos asociados al desarrollo del cáncer humano, así como en la restitución epitelial [36].

Por ejemplo, en un estudio realizado en 2002 se observó que este biomarcador se expresaba con frecuencia en tumores de mama, desempeñando un papel en la movilidad y diseminación de las células cancerígenas. En particular, se le atribuyen funciones como la estimulación del movimiento celular, un proceso clave en fenómenos como la invasión y la metástasis, además de actuar como mediador de los estrógenos [37].

De manera similar, investigaciones más recientes han encontrado su asociación con ciertos subtipos de retinoblastoma, un cáncer ocular común en niños, donde se presenta sobreexpresado en casos específicos de la enfermedad [38].

En conjunto, la evidencia encontrada respalda su relación con distintos tipos de cáncer, lo que constituye un argumento sólido para considerarlo como una variable relevante dentro del modelo, dado su posible aporte en la diferenciación entre pacientes con y sin cáncer.

Sin embargo, este tipo de análisis podría no ser suficiente, dado que se dispone de un conjunto de datos limitado. Al tratarse de una muestra pequeña, existe el riesgo de que no represente adecuadamente la realidad. Por esta razón, y con el fin de respaldar la decisión final, se examinó nuevamente la distribución de la variable en el conjunto de datos disponible. Para ello, se elaboró un diagrama de cajas (Figura 8). En este se observa que la mediana de los

niveles de TFF1 es considerablemente más alta en los casos positivos (diagnóstico = 1) en comparación con los negativos (diagnóstico = 0). Asimismo, el rango intercuartílico de la clase positiva es más amplio, lo que indica una mayor dispersión. Esta clase también presenta valores atípicos significativamente más altos, alcanzando niveles muy superiores a los observados en la clase negativa. Por otro lado, los casos negativos muestran un rango intercuartílico más estrecho y concentrado en valores bajos, lo cual indica una menor variabilidad y una alta concentración de bajos niveles de TFF1. Esto implica que al menos el 75 % de los casos sin cáncer presentan niveles reducidos de esta variable, con una superposición muy baja respecto a los casos con cáncer. En general, estos hallazgos evidencian una clara separación entre ambas clases, aunque no completa, lo que refuerza el potencial discriminativo de TFF1. En conclusión, tanto la revisión de la literatura sobre la naturaleza de TFF1 como el análisis de su distribución en los datos disponibles respaldan su relevancia como variable predictiva. Esto permitió justificar su inclusión en el proceso de entrenamiento de los modelos, ya que, en combinación con otras características, puede contribuir

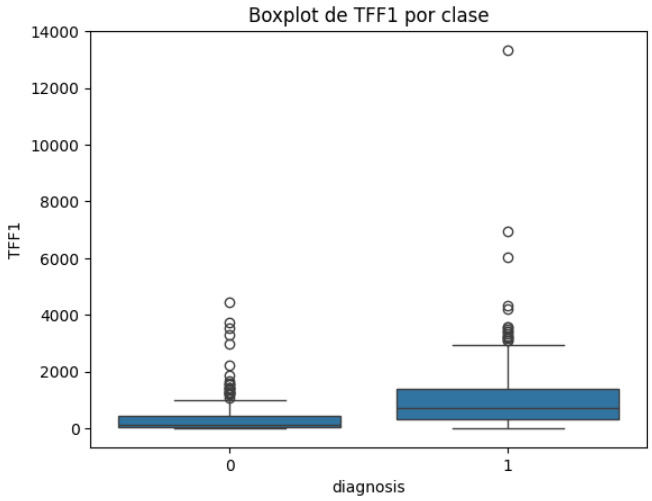

Figura 8 Boxplot TFF1/diagnóstico

significativamente a mejorar la capacidad de clasificación.

La siguiente característica evaluada fue REG1A. No se encontró abundante información sobre su papel específico en el desarrollo del cáncer en humanos, sin embargo, sí está claramente asociado con lesiones pancreáticas. En particular, se relaciona con la inflamación y la regeneración del tejido pancreático, por lo que su expresión puede incrementarse significativamente tras una lesión [19].

Inicialmente, se consideró que por haber agrupado dos clases (incluyendo enfermedades hepatobiliares como la pancreatitis) la inclusión de REG1A podría resultar contraproducente dado que REG1A también está estrechamente vinculado a la pancreatitis. De hecho, diversos estudios sugieren que este biomarcador puede ser útil tanto para el diagnóstico como para diferenciar tipos de pancreatitis e incluso predecir su progresión [19].

No obstante, al analizar la distribución por clase mediante un diagrama de cajas (Figura 9), se observó una ligera diferencia que podría resultar relevante al combinarse con otras características en el proceso de clasificación. Es importante señalar que esta variable requirió imputación de valores nulos utilizando la mediana, lo cual explica la notable presencia de valores atípicos en el gráfico.

A pesar de ello, se aprecia una diferencia entre ambas clases: el rango intercuartílico de los casos positivos (diagnóstico 1) es más amplio y se encuentra desplazado hacia valores superiores. Esto indica que los valores centrales de REG1A tienden a ser más altos en comparación con los casos negativos (diagnóstico 0). Este comportamiento podría sugerir que, en el desarrollo del cáncer (como el PDAC), el páncreas experimenta un mayor grado de lesión e inflamación que en condiciones como la pancreatitis, lo que justificaría niveles más elevados de REG1A. En síntesis, aunque la evidencia teórica no era completamente concluyente para respaldar el uso de este biomarcador en la clasificación, el análisis empírico de su distribución dentro del conjunto de datos motivó su inclusión en el modelo.

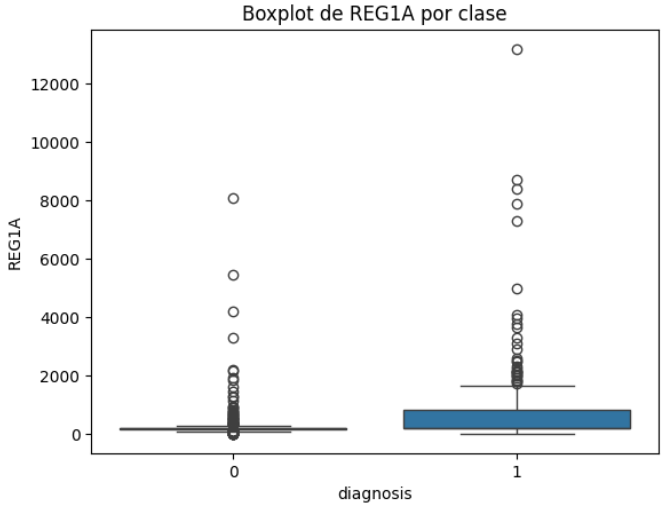

Figura 9 Boxplot REG1A/diagnóstico

Posteriormente, al revisar la literatura relacionada con el siguiente biomarcador, REG1B, se encontró que este está estrechamente involucrado en el desarrollo del cáncer de páncreas. Diversos estudios indican que se encuentra significativamente sobreexpresado en casos diagnosticados, y aún más en aquellos que llevan de 3 a 6 años con la enfermedad. Además, se ha reportado que presenta efectos causales en la progresión del cáncer pancreático [39]. De hecho, en dicho estudio se evidenció que este biomarcador puede encontrarse elevado en sangre varios años antes de la aparición del cáncer, lo que sugiere su potencial como indicador temprano, incluso antes de que la enfermedad alcance una etapa invasiva.

Debido a esta fuerte relación con el cáncer de páncreas, se consideró necesario analizar su distribución en el conjunto de datos antes de incluirlo como característica en el entrenamiento de los modelos. Para ello, se utilizó un diagrama de cajas (Figura 10), a partir del cual se identificaron patrones relevantes.

En primer lugar, se observa la presencia de numerosos valores atípicos en ambas clases; sin embargo, estos son considerablemente más altos y frecuentes en los casos con diagnóstico positivo (diagnosis = 1). Asimismo, la mediana de REG1B es claramente superior en la clase positiva, lo que indica una mayor expresión del biomarcador en estos casos. Por otro lado, el rango intercuartílico de la clase negativa (diagnosis = 0) es estrecho y se concentra en valores bajos, lo que evidencia una baja variabilidad y una alta concentración de observaciones con niveles reducidos del biomarcador. En contraste, la clase positiva presenta un rango intercuartílico más amplio y desplazado hacia valores más altos, lo que indica una mayor dispersión y una tendencia general a niveles elevados de REG1B. Aunque existe cierto grado de solapamiento entre ambas clases, este es muy poco, lo que sugiere una capacidad discriminativa relevante. En conclusión, tanto la evidencia encontrada en la literatura como el análisis de la distribución de REG1B respaldan su importancia como variable predictiva. Por ello, se decidió incluir este biomarcador en el conjunto de características para el entrenamiento de los modelos, considerando que, en conjunto con las demás variables seleccionadas, puede contribuir significativamente a mejorar el desempeño en la clasificación.

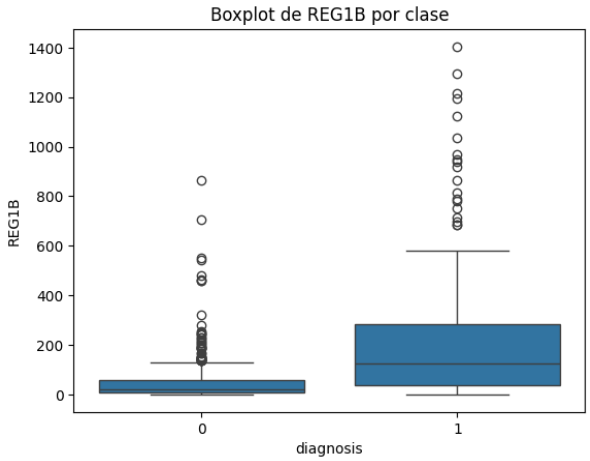

Figura 10 Boxplot REG1B/diagnóstico

Posteriormente, se evaluó la variable creatinina. En este caso, se decidió prescindir de una revisión exhaustiva de la literatura sobre su posible relación con el cáncer de páncreas, ya que el análisis exploratorio de los datos proporcionaba evidencia suficiente para la toma de decisiones.

A partir del diagrama de cajas (Figura 11), se observa que ambas clases (diagnosis = 0 y diagnosis = 1) presentan distribuciones muy similares. En particular, las medianas de ambas clases son prácticamente iguales, lo que indica que no existe una diferencia significativa en los valores centrales de la creatinina entre individuos con y sin diagnóstico positivo. Asimismo, los rangos intercuartílicos son muy parecidos, con una ligera elevación en la clase positiva, sin embargo, este desplazamiento es mínimo y no resulta suficiente para establecer una separación clara entre las clases. Además, se evidencia un solapamiento casi total entre ambas distribuciones, lo que sugiere una baja capacidad discriminativa de esta variable. Si bien se observan algunos valores atípicos más altos en la clase positiva, estos son pocos y no modifican

de manera sustancial la tendencia general de los datos. En conjunto, la alta similitud en la distribución, el solapamiento de los rangos intercuartílicos y la cercanía de las medianas indican que la creatinina no aporta información relevante para diferenciar entre las clases. En consecuencia, se concluyó que la inclusión de esta variable en el entrenamiento de los modelos podría no solo ser innecesaria, sino potencialmente contraproducente, al introducir ruido o información poco discriminativa. Por ello, se decidió excluir la creatinina del conjunto de características, considerando que no contribuiría de manera significativa a la identificación de casos con cáncer de páncreas frente a aquellos sin la enfermedad.

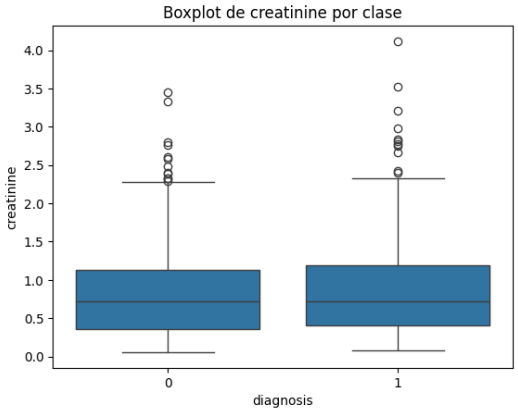

Figura 11 Boxplot creatinina/diagnóstico

La última característica por evaluar fue el biomarcador Plasma CA19-9, el cual finalmente se incluyó dentro del conjunto de variables utilizadas para el entrenamiento de los modelos.

En primer lugar, se analizó su distribución por clase mediante un diagrama de cajas y bigotes (Figura 12). En esta gráfica se observa una gran cantidad de valores atípicos, especialmente en la clase con diagnóstico positivo (diagnosis = 1). Cabe destacar que esta fue una de las variables en las que se imputaron valores faltantes utilizando la mediana, lo cual puede contribuir a la presencia de ciertos valores extremos. No obstante, también es importante considerar que niveles elevados de CA19-9 suelen estar asociados con etapas más avanzadas de la enfermedad, lo que explica la aparición de valores considerablemente más altos en la clase positiva [40]. Por otro lado, los datos correspondientes a la clase negativa (diagnosis = 0) se encuentran fuertemente concentrados en valores muy bajos, hasta el punto de que el rango intercuartílico resulta casi imperceptible en la gráfica. Esto indica una muy baja variabilidad en los niveles de este biomarcador para individuos sin la enfermedad. En contraste, aunque en la clase positiva también existe concentración en valores bajos, la mediana se encuentra un poco desplazada hacia valores superiores, y el rango intercuartílico es más amplio, lo que evidencia una mayor dispersión y una tendencia general a valores más elevados. Si bien existe cierto grado de solapamiento entre ambas clases en los valores bajos, la diferencia en la dispersión y, especialmente, la presencia de valores extremos significativamente altos en la clase positiva sugiere un alto potencial discriminativo. Este patrón indica que, aunque no todos los casos positivos presentan niveles elevados, aquellos que sí lo hacen pueden ser claramente distinguibles de los casos negativos.

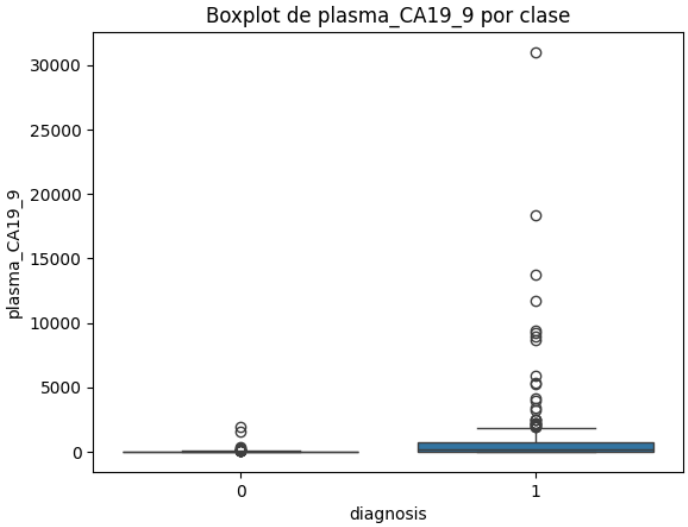

Figura 12 Boxplot CA19-9/diagnóstico

Con el fin de respaldar esta observación, se realizó una revisión de estudios sobre el CA19-9 y su relación con el páncreas, o el desarrollo del cáncer. Se encontró que este biomarcador es uno de los más utilizados en la práctica clínica, tanto para el diagnóstico como para el monitoreo del tratamiento en pacientes con cáncer de páncreas (PDAC) [40]. Además, se ha reportado su elevación en otras patologías, como la ictericia obstructiva, la pancreatitis y ciertos tipos de cáncer (mama, hígado, colon-recto y estómago) [39]. La sobreexpresión de CA19-9 en el cáncer de páncreas se asocia con alteraciones en los procesos de glicosilación durante la transformación maligna de las células. Además, se ha observado que pacientes con niveles normales de este biomarcador tienden a presentar menor invasión de tejidos circundantes y menor compromiso ganglionar, mientras que niveles elevados se relacionan con una mayor extensión tumoral y con invasión linfovascular [40]. En conclusión, tanto el análisis de la distribución en los datos como la evidencia reportada en la literatura respaldan la inclusión del CA19-9 como una variable relevante. Su capacidad para reflejar la progresión de la enfermedad y su comportamiento diferenciado entre clases lo convierten en una característica valiosa que, en conjunto con otras variables, puede contribuir significativamente al desempeño de los modelos de clasificación.

3.2.4. Normalización de características

Por otra parte, dada la naturaleza de los datos y la distribución observada, se decidió aplicar un proceso de normalización de las características. El objetivo principal de esta etapa fue adecuar las variables para el entrenamiento de los modelos, evitando que aquellas con valores más grandes dominaran sobre las demás y afectaran el aprendizaje. De esta forma, se buscó llevar todas las variables a escalas comparables, lo que facilita la convergencia de algunos algoritmos y mejora la estabilidad de los resultados. Esto resulta especialmente relevante al trabajar con biomarcadores, cuyos rangos pueden diferir considerablemente.

Para ello, se emplearon dos técnicas disponibles en la librería scikit-learn de Python. La primera fue MaxAbsScaler, seleccionada debido a que está diseñada para escalar datos con

diferentes magnitudes sin alterar su estructura. Esta elección fue pertinente, ya que los biomarcadores analizados presentan rangos muy distintos: por ejemplo, mientras el CA19-9 puede tomar valores entre 0 y 31.000, LYVE1 varía aproximadamente entre 0,001 y 23,89. Esta diferencia podría otorgar una mayor influencia a variables con valores más altos, como CA19-9, sin que necesariamente sean más relevantes en términos predictivos.

La segunda técnica utilizada fue RobustScaler, recomendada cuando existen numerosos valores atípicos. Este método se basa en estadísticas robustas, como la mediana y el rango intercuartílico, para reducir la influencia de valores extremos. Su uso se justificó a partir del análisis previo, donde se evidenció una alta presencia de valores atípicos en varias características. Estos valores, además de ser numerosos, presentaban magnitudes elevadas respecto a la tendencia general, lo que podría afectar negativamente el desempeño de algunos modelos y generar sesgos en el aprendizaje.

Finalmente, se optó por trabajar con tres versiones del conjunto de datos: una sin normalización (únicamente con el tratamiento de valores nulos), otra escalada mediante MaxAbsScaler y una tercera utilizando RobustScaler. Esta estrategia permitió comparar el desempeño de los modelos bajo diferentes condiciones de preprocesamiento, con el fin de identificar cuál de ellas ofrecía mejores resultados y se ajustaba de manera más adecuada al problema planteado.

3.3. ENTRENAMIENTO DE LOS MODELOS

Primero, es importante mencionar cada uno de los modelos de aprendizaje automático que se decidió utilizar, los cuales fueron 5 y se eligieron de forma arbitraria, es decir que no hay una razón lógica o conceptual por la cual se generó una inclinación hacia ellos, sino, al contrario, fue por voluntad y porque ya se conocían con anterioridad. Los 5 modelos elegidos fueron: Máquina de Vectores de Soporte (SVM), XGBoost, K vecinos más cercanos (KNN), Random Forest y Regresión Lineal.

3.3.1. GridSearch

El siguiente paso consistió en aplicar GridSearch a cada uno de los modelos de aprendizaje automático implementados. Esta técnica permite realizar una búsqueda exhaustiva sobre un conjunto de valores predefinidos para los hiperparámetros de un estimador. En otras palabras, se le proporciona al algoritmo una lista de posibles configuraciones junto con el modelo a evaluar, y este se encarga de probar todas las combinaciones posibles. Como resultado, GridSearch identifica aquella combinación de hiperparámetros que ofrece el mejor desempeño según la métrica de evaluación definida, lo que permite mejorar el rendimiento del modelo de manera sistemática y objetiva.

Primero, antes de pasar a cada una de las listas de hiperparametros que se usaron en cada uno de los algoritmos de aprendizaje, es importante aclarar que cada lista de parámetros se eligió arbitrariamente, es decir que no hay una razón especifica o conceptual por la que se eligió cada uno de los parámetros, sino que fue una elección voluntaria observando la función de cada uno de ellos en la documentación oficial de Scikit-learn. Otra observación que es importante realizar, es que para cada algoritmo que tenía un parámetro del tipo

"class_weight", se usó una sola posibilidad, es decir que siempre en este parámetro (que asigna un peso a cada clase) se pasó el mismo valor: 1 para la clase negativa, y 2 para la clase positiva. Esto le daba el doble de peso a la clase positiva frente a la negativa, dado que se consideró que era más importante acertar en la clasificación de quienes tienen la enfermedad que clasificar correctamente a quienes no la padecen.

El primer algoritmo al que se le aplicó la búsqueda de hiperparámetros fue SVM. Para ello, inicialmente se definió la siguiente red de parámetros:

- En el parámetro "kernel", que especifica el tipo de función utilizada por el algoritmo, se consideraron los valores "linear", "rbf" y "poly".
- En el parámetro "C", correspondiente al término de regularización, se evaluaron los valores 5 y 10.
- En el parámetro "degree", utilizado en kernels polinómicos, se incluyeron los valores 2, 3, 4 y 5.

Sin embargo, esta configuración generó un número elevado de combinaciones, lo que provocó que la ejecución de GridSearch superara las 12 horas sin completarse. Debido al costo computacional tan alto, se decidió simplificar la red de búsqueda eliminando el parámetro "degree", reduciendo así significativamente el número de combinaciones a evaluar. De este modo, la búsqueda se limitó a los parámetros "kernel" y "C". Una vez finalizada la ejecución, GridSearch determinó que la mejor combinación correspondía al uso del kernel "linear" junto con un valor de 5 para el parámetro "C", lo que permitió obtener el mejor desempeño del modelo bajo las condiciones evaluadas.

El siguiente algoritmo fue K Nearest Neighbors (KNN), para este algoritmo se usó una red de parámetros un poco más extensa, la cual se definió de la siguiente manera:

- Para el parámetro que define el número de vecinos, etiquetado como "n_neighbors", se pasó una lista con los valores 3, 5, 7 y 9.
- Para aquel que define el algoritmo utilizado para calcular los vecinos más cercanos, "algorithm", se evaluaron "ball_tree" y "kd_tree".
- En el parámetro que establece la función de ponderación usada en la predicción, nombrado como "weights" se evaluaron "uniform" y "distance".
- Por último, en "p", se consideraron los valores 1, lo cual equivale a usar la distancia de Manhattan entre los vecinos, y 2 que equivale a usar la distancia euclidiana.

En esta búsqueda, la función GridSearch, estableció que, de los valores establecidos para los hiperparametros, la mejor combinación eran los valores 7, "ball_tree", "uniform" y 2, respectivamente, por lo que esta fue la lista usada para la evaluación de los modelos.

Después, se evaluó una red de parámetros para el algoritmo regresión logística, la cual se construyó así:

- En "penalty" que define la norma de la penalización, se usaron los valores "l1", "l2", y None.
- Para "solver", parámetro en el que se especifica el algoritmo a utilizar en el problema de optimización, se usaron "lbfgs", "newton-cholesky", "newton-cg", "liblinear", y

"saga". Aquí no todos los valores son compatibles con todos los valores en "penalty", aun así, GridSearch únicamente devuelve una advertencia mostrando la incompatibilidad, sin interrumpir la ejecución.
- Para el parámetro "C", que proporciona el inverso de la fuerza de regularización (valores más pequeños especifican una regularización más fuerte), se usaron los valores 1, 5, 10.
- Para "max_iter", el cual indica el número máximo de iteraciones necesarias para que los solucionadores converjan, se enlistaron los valores 100, 500, 1000.
- Por último, como ya se especificó, en "weight_class", que son los pesos asociados a las clases, se adjuntaron los valores 1 para la clase negativa, y 2 para la clase positiva, que indica el doble de peso para la segunda clase con respecto a la primera.

El resultado arrojado por la función de búsqueda en esta ocasión determinaba que la mejor combinación de parámetros para el algoritmo era "l2", "lbfgs", 1, y 1000, respectivamente.

El cuarto algoritmo evaluado fue XGBoost, para el cual se construyó la siguiente lista de parámetros para realizar la búsqueda con GridSearch:

- En el parámetro "n_estimators", que establece el número de rondas de Boosting, se usaron los valores 1, 5 y 10.
- El siguiente parámetro evaluado fue "grow_policy", que se usa para especificar la política de crecimiento de árboles, se enlistaron "depthwise" y "lossguide".
- Para aquel parámetro que indica la profundidad máxima del árbol para los aprendices base, etiquetado como "max_depth", se evaluaron los valores 1, 2 y 3.
- Por último, en "booster", que se usa para determinar que potenciador se usará, se presentaron los valores "gbtree", "gblinear" y "dart".

La combinación óptima que se obtuvo según la función de búsqueda utilizada fue 10, "depthwise", 2 y "gbtree".

Por último, la red de parámetros usada para evaluar su combinación en el algoritmo Random Forest fue:

- En "n_estimators", que establece el número de árboles en el bosque se listaron los valores 100, 150, 175, 190, 200.
- Para el parámetro "criterion" en el que se especifica la función para medir la calidad de una división, se evaluaron "gini", "entropy" y "log_gloss".
- Para "min_samples_split" que indica el número mínimo de muestras necesarias para dividir un nodo se usaron 2, 3 y 4.

Para esta lista de parámetros, el resultado obtenido por GridSearch como la mejor combinación con los valores proporcionados, fue de 175 estimadores, "gini" para "criterion", y 2 muestras mínimo para dividir un nodo.

Una vez definido un conjunto óptimo de hiperparámetros para cada modelo, garantizando el mejor desempeño dentro de las combinaciones evaluadas, se procedió a la fase de entrenamiento y prueba. Para ello, se emplearon diversas técnicas de validación, junto con la evaluación del rendimiento mediante seis métricas diferentes. Esto permitió realizar una

comparación más completa y objetiva entre los modelos, facilitando la selección de aquel que ofreciera los mejores resultados.

La explicación de la utilidad o función de cada uno de los parámetros se obtuvo de la documentación oficial de Scikit-Learn [30] *para todos los algoritmos menos para XGBoost, la cual se obtuvo de la documentación oficial de XGBoost* [41].

3.3.2. Métricas de evaluación

Existen múltiples métricas y métodos para evaluar el rendimiento de los modelos de aprendizaje automático. Es fundamental emplearlas durante la evaluación, ya que permiten cuantificar el desempeño del modelo. Estas métricas proporcionan información clave sobre qué tan bien el modelo identifica patrones en los datos y cómo se comporta frente a nuevas observaciones. Además, facilitan la comparación entre distintos modelos o configuraciones, lo que permite seleccionar la alternativa más adecuada para la clasificación o predicción de la presencia o ausencia de cáncer a partir de las características analizadas.

Una de las métricas utilizadas fue la exactitud (accuracy), la cual representa la proporción de muestras clasificadas correctamente. Esta métrica permite comparar de forma sencilla el rendimiento de los modelos basado en la precisión de sus predicciones durante la fase de prueba. Sin embargo, utilizar únicamente la exactitud no es suficiente para evaluar el desempeño de los modelos. Por ejemplo, en conjuntos de datos desbalanceados, donde una clase es más frecuente que otra, un modelo podría obtener una alta exactitud simplemente clasificando correctamente la mayoría de los casos de la clase predominante, sin necesariamente aprender a distinguir adecuadamente la clase minoritaria. Por esta razón, la exactitud se complementó con otras métricas que permiten analizar el rendimiento del modelo desde diferentes perspectivas, considerando no solo los aciertos en general, sino también el comportamiento frente a cada una de las clases y los distintos tipos de error.

Una métrica utilizada para acompañar la exactitud fue la sensibilidad, que mide la fracción de muestras positivas clasificadas correctamente durante la prueba. Esta fue útil para medir que proporción de las muestras con presencia de cáncer fueron clasificadas por el modelo realmente como muestras con presencia de cáncer.

Otra métrica fue la especificidad que, de manera similar a la métrica anterior, mide la fracción de muestras negativas clasificadas correctamente, es decir que muestra qué cantidad de las muestras con ausencia de cáncer pancreático fueron clasificadas realmente en esta categoría por el modelo.

La siguiente métrica fue la precisión, esta mide qué fracción de las muestras que fueron clasificadas como positivas por el modelo son realmente muestras positivas, es decir, que mide cuales de las muestras que el modelo clasificó como “con presencia de cáncer” entraban en realidad dentro de esta clase.

Otra métrica relevante es el F1-score, la cual combina la precisión y la sensibilidad (recall) en una sola medida. Esta métrica se calcula como la media armónica entre ambas, lo que implica que solo toma valores altos cuando tanto la precisión como la sensibilidad son elevadas. En otras palabras, el F1-score evalúa el equilibrio entre la capacidad del modelo para identificar

correctamente los casos positivos y su capacidad para evitar falsos positivos. El F1-score es especialmente útil en contextos donde existe desbalance entre clases, ya que no se ve influenciado únicamente por la cantidad de aciertos globales, sino que penaliza tanto los falsos positivos como los falsos negativos. En este sentido, esta métrica permite tener una visión más completa del desempeño del modelo, mostrando qué tan bien logra detectar los casos de cáncer sin clasificar incorrectamente a pacientes sanos como positivos.

La última métrica utilizada para evaluar a los modelos fue el AUC-ROC o área bajo la curva ROC, la cual tiene una interpretación probabilística, esta mide la probabilidad de que una muestra positiva tenga una puntuación de clasificación más alta como positiva que una negativa. Un valor cercano a 1 indica que el modelo tiene una alta capacidad para diferenciar correctamente entre pacientes con y sin cáncer, mientras que un valor cercano a 0.5 sugiere un comportamiento similar al azar. Por lo general, el modelo con mayor área debajo de la curva es el mejor.

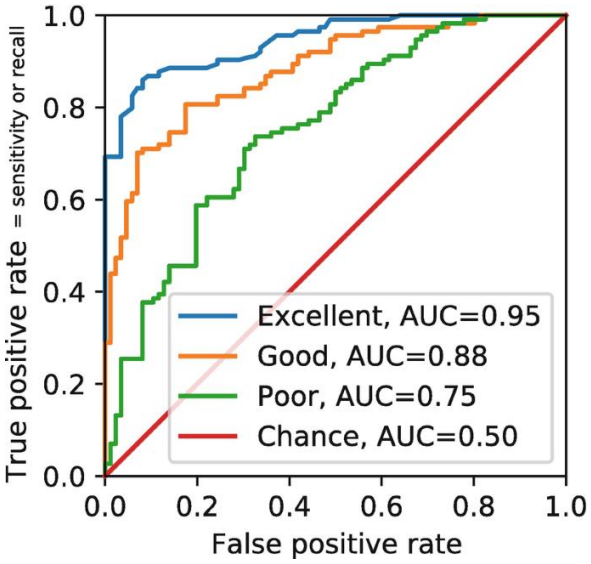

Figura 13 Curva ROC para diferentes clasificadores [23].

3.3.3. Técnicas de validación cruzada

Además de la división típica en subconjuntos de entrenamiento y prueba (train-test), las técnicas de validación cruzada se utilizan para obtener una evaluación más robusta y confiable del desempeño de los modelos de aprendizaje automático. Estas metodologías permiten entrenar y evaluar el modelo múltiples veces sobre diferentes divisiones de los datos, lo cual reduce la dependencia de una única división y disminuye el riesgo de obtener resultados sesgados o poco representativos. De esta manera, la validación cruzada proporciona una estimación más estable de la capacidad de clasificación del modelo ante datos desconocidos, lo que resulta especialmente importante cuando se trabaja con conjuntos de datos limitados, ya que permite aprovechar mejor la información disponible y tomar decisiones más informadas al comparar modelos.

Para evaluar cada uno de los modelos se emplearon tres técnicas de validación. La primera de ellas fue Leave One Out (LOOCV), la cual consiste en que, dado un conjunto de datos con n observaciones, el modelo se entrena n veces. En cada iteración se utilizan n-1 datos para el entrenamiento y el dato restante para la prueba, de modo que cada dato se usa una única vez para la validación. Aquí, el resultado de cada iteración es binario: el modelo acierta completamente o falla en la predicción de ese único dato. Esta técnica tiene como principal ventaja que maximiza el tamaño del conjunto de entrenamiento en cada iteración, lo cual resulta especialmente útil en conjuntos de datos pequeños, donde es importante aprovechar al máximo la información disponible. Además, permite simular de manera más precisa el comportamiento del modelo ante la llegada de nuevos datos. Sin embargo, también puede implicar un mayor costo computacional y una mayor variabilidad en los resultados.

Otra técnica utilizada fue Hold-Out, la cual consiste en dividir el conjunto de datos en dos partes: un subconjunto para el entrenamiento y otro para la prueba. Esta estrategia, aunque es sencilla y ampliamente utilizada, puede presentar desventajas cuando se trabaja con conjuntos de datos limitados, ya que una sola partición puede no ser lo suficientemente representativa y generar una alta variabilidad en los resultados. En la validación, se empleó una división de 75%-25% definida de forma arbitraria, donde el 75% de los datos se destinó al entrenamiento y el 25% a la prueba. Además, el proceso se repitió 10 veces, realizando en cada iteración una partición aleatoria diferente. Esto permitió obtener una evaluación más estable y confiable del desempeño de los modelos, al considerar múltiples configuraciones de los datos. Para medir el rendimiento se calcularon las métricas en cada una de las iteraciones y al finalizar se obtuvo tanto el promedio como la desviación estándar.

Por último, se empleó la técnica de validación K-Fold, la cual consiste en dividir el conjunto de datos en K subconjuntos o particiones. El modelo se entrena K veces, utilizando en cada iteración K-1 subconjuntos para el entrenamiento y el subconjunto restante para la prueba, de manera que cada partición se utiliza como conjunto de validación una única vez. En este caso, se definió un valor de K = 10. Durante cada iteración se calcularon las métricas de rendimiento, y al finalizar el proceso se obtuvo el promedio de dichas métricas junto con su desviación estándar. Esto permitió no solo evaluar el desempeño general del modelo, sino también medir la variabilidad de los resultados, proporcionando una estimación más robusta y confiable de su capacidad de generalización.

3.4. INTERFAZ GRÁFICA DE USUARIO

Uno de los objetivos del proyecto fue el desarrollo de una interfaz gráfica que facilitara a los usuarios el uso del modelo para realizar predicciones a partir de sus propios biomarcadores o los de otra persona. Para ello, se diseñó una interfaz sencilla y de uso intuitivo, basada en un formulario que, al ser completado y enviado, permite obtener un resultado que indica si, según el modelo, el caso corresponde a una clasificación positiva (presencia de cáncer pancreático) o negativa (ausencia de la enfermedad). Esta interfaz se conecta con el modelo seleccionado, el cual fue elegido por presentar el mejor desempeño entre los cinco evaluados durante el proceso. Posteriormente, se decidió adicionar una sección que muestra algunas de las

métricas de evaluación del modelo, con el fin de evidenciar su rendimiento y aportar mayor transparencia y confianza al usuario respecto a la predicción obtenida.

Conviene subrayar que durante el desarrollo se tuvo especial cuidado con la información presentada y la forma en que se comunican los resultados, dado que este tipo de herramientas en contextos médicos puede dar lugar a interpretaciones inadecuadas. En particular, se buscó evitar que el usuario asuma que el modelo ofrece diagnósticos definitivos o que puede reemplazar la evaluación de un profesional de la salud. Una interpretación errónea podría generar alarma innecesaria ante un resultado positivo o, por el contrario, llevar a descartar la consulta médica ante un resultado negativo, incluso cuando existan síntomas. Por esta razón, la interfaz fue concebida como una herramienta de apoyo y no como un sustituto del diagnóstico clínico.

3.4.1. Diagramas de actividades.

Un diagrama de actividades es una representación gráfica enfocado en el flujo de acciones y pasos dentro de un sistema. Este tipo de diagrama permite visualizar diferentes tareas, decisiones y caminos que sigue un proceso desde su inicio hasta su final. Estos diagramas son muy relevantes en el desarrollo de software para entender como interactúan los componentes y para diseñar y definir la lógica de funcionamiento antes de realizar la implementación. Así, es posible tener una mejor planificación y diseño del sistema.

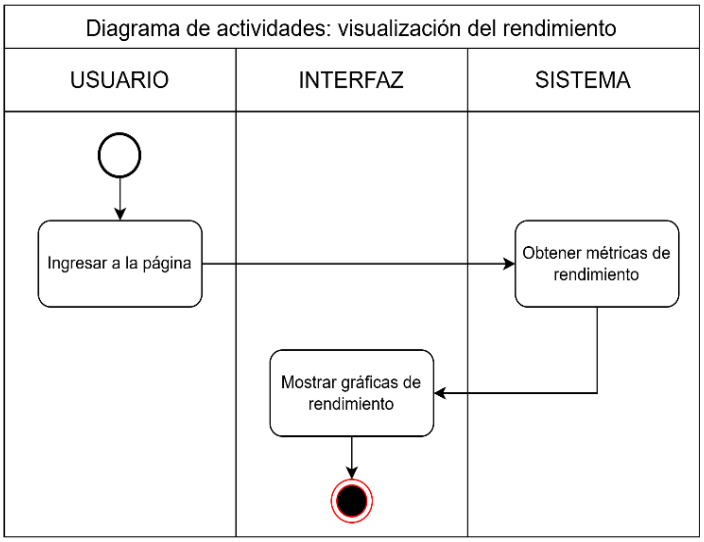

Figura 14 Diagrama de actividades: resultados

Dado que se deseaba implementar dos secciones, se inició con el diseño del diagrama de actividades correspondiente a la sección de resultados (Figura 14). Esta sección, como se mencionó anteriormente, tiene como finalidad mostrar al usuario las métricas obtenidas por el modelo durante su evaluación. Además, corresponde a la vista principal de la aplicación, es decir, la primera que se presenta al acceder a la página. Al momento de la carga, se asume que el modelo ya ha sido previamente entrenado y evaluado, por lo que la interfaz únicamente se comunica con el sistema para solicitar las métricas de desempeño. Una vez obtenidas, estas

se procesan y se muestran al usuario de manera clara. Por lo tanto, el flujo de actividades se centra en la solicitud de información, la recepción de los resultados y su visualización, por lo que el diagrama de actividades se diseñó en torno a estos pasos.

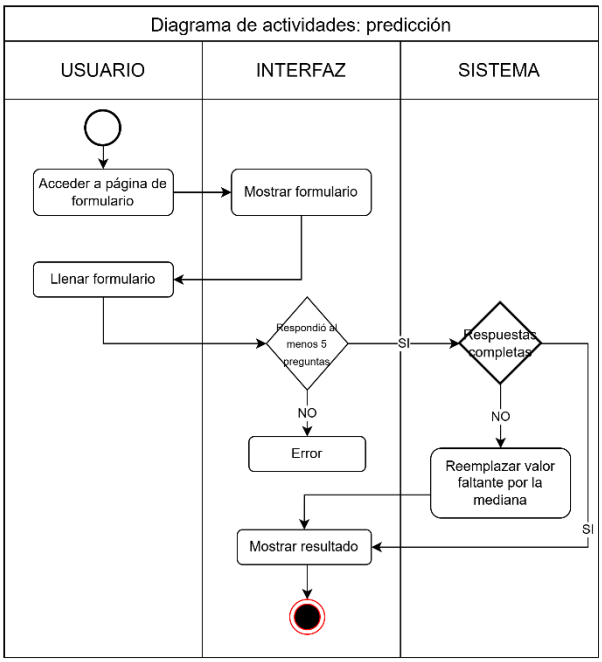

Figura 15 Diagrama de actividades: formulario de predicción.

El siguiente diagrama de actividades describe el flujo de acciones necesario para obtener un resultado a partir del formulario (Figura 15). Aunque es más detallado que el anterior, sigue siendo un diagrama sencillo, acorde con la simplicidad de la interfaz. En este caso, al ingresar a la sección correspondiente, la interfaz muestra el formulario al usuario. Una vez diligenciado y enviado, el sistema realiza una validación sobre la cantidad de campos completados. Si el usuario no responde al menos 5 de las 6 preguntas, se genera un mensaje de error indicando que es necesario contar con un mínimo de cinco características. Esta restricción se establece porque no sería apropiado realizar la predicción con menos datos de los utilizados en el entrenamiento ya que el modelo fue entrenado para identificar patrones en un conjunto especifico de características, por lo que, al proporcionar una cantidad menor, se puede reducir significativamente la calidad de la predicción además de que se aleja del caso real del usuario. Por otro lado, si falta únicamente una característica (distinta de la edad), el sistema sustituye el valor faltante utilizando la mediana calculada a partir de los datos de entrenamiento. Posteriormente, con la información completa, se realiza la predicción en el modelo y el resultado es enviado a la interfaz, la cual se encarga de presentarlo al usuario de forma clara.

Contar con una planificación más estructurada facilita la construcción del sistema, ya que permite tener una idea clara y concreta de lo que se pretende desarrollar. Esto hace que el proceso sea óptimo y organizado, evitando divagaciones durante el desarrollo que puedan incrementar innecesariamente la complejidad de esta etapa.

3.4.2. Mockup.

Un mockup es una representación visual que muestra el aspecto final de un diseño, en la cual se incluyen elementos como colores, tipografías e imágenes. Este tipo de recurso permite obtener una vista previa de cómo se presentará un sitio web o una aplicación ante los usuarios [42]. Los mockups son importantes por varias razones. En primer lugar, facilitan la comunicación de las ideas de diseño entre las partes involucradas, lo que permite realizar ajustes y correcciones antes de iniciar el desarrollo del producto final. En segundo lugar, sirven como una guía para los desarrolladores, ya que proporcionan una base clara sobre la cual trabajar, haciendo el proceso más organizado. En este caso, se decidió elaborar un mockup con el fin de plasmar una idea del diseño final de la interfaz y facilitar el desarrollo del sistema. De esta manera, se evitó incrementar la complejidad del proceso al separar la etapa de diseño de la implementación, permitiendo un flujo de trabajo más ordenado y eficiente.

Luego, para la elaboración del mockup se utilizó la herramienta Canva. Aunque existen herramientas más especializadas para este tipo de diseño, se optó por esta debido a su facilidad de uso y a que no era necesario desarrollar un mockup complejo o altamente detallado, considerando la simplicidad de la interfaz. Por lo tanto, al ser una herramienta conocida y fácil de usar y que no requería de tiempo de aprendizaje fue fundamental para inclinarse por Canva para el diseño de este.

Primero, se realizó el diseño de la sección que expone los resultados de las métricas utilizadas para evaluar al modelo (Figura 16), para eso se usó un gráfico de dona para cada una de las características indicando el porcentaje obtenido en dicha métrica al momento de la evaluación.

Figura 16 Mockup: sección resultados.

En segundo lugar, se realizó el diseño de la sección del formulario (Figura 17) en la que se muestra una entrada para cada una de las características elegidas anteriormente en el proceso de entrenamiento para diligenciar por el usuario. En esta misma sección se expone una advertencia para aclarar que este sistema no muestra un resultado definitivo. Además, en el

diseño se incluyeron las ventanas que muestran cada uno de los dos resultados posibles, ausencia de cáncer de páncreas (Figura 19) y presencia de este (Figura 18).

Figura 17 Mockup: sección del formulario.

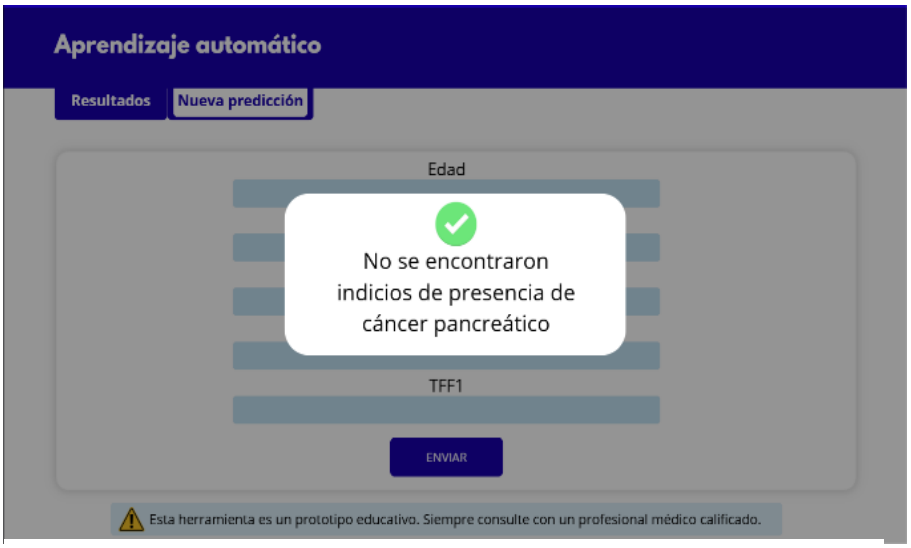

Figura 18 Mockup: presencia de cáncer.

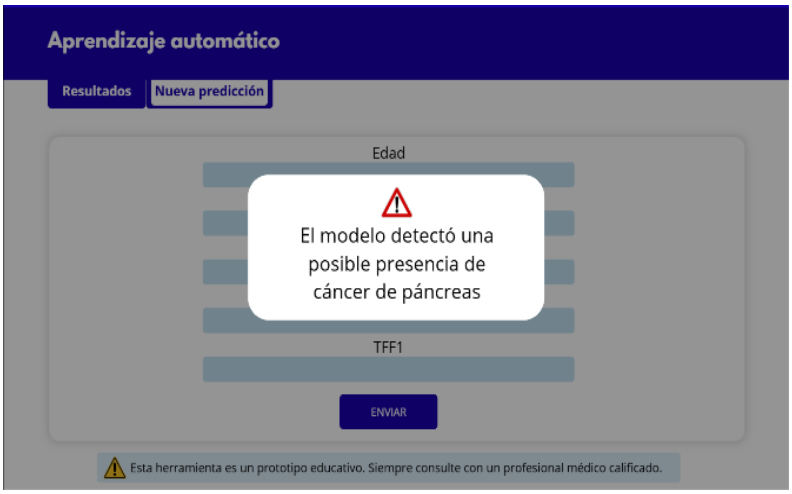

Figura 19 Mockup: ausencia de cáncer.

Una vez finalizado el mockup y definida una idea concreta del diseño, se procedió al desarrollo de la interfaz. En esta etapa se siguieron tanto la paleta de colores como la idea general planteada previamente, abarcando las distintas secciones del sistema: la visualización de métricas, el formulario de entrada de datos y la presentación de los resultados. Esto permitió mantener coherencia entre el diseño inicial y la implementación final, facilitando un desarrollo más ordenado y alineado con los objetivos del proyecto.

4. RESULTADOS

A continuación, se presentan los resultados obtenidos durante la fase de entrenamiento y prueba de los cinco modelos seleccionados. Para su evaluación, como se ha mencionado previamente, se emplearon seis métricas de rendimiento aplicadas en tres métodos de validación, lo que permitió realizar una comparación del desempeño de cada modelo y facilitar la selección del más adecuado. Adicionalmente, el conjunto de datos fue normalizado utilizando dos técnicas diferentes proporcionadas por la librería Scikit-learn, por lo que cada modelo fue entrenado y evaluado sobre tres versiones del dataset: sin normalización, con MaxAbsScaler y con RobustScaler. Esto permitió analizar el impacto del preprocesamiento en los resultados obtenidos. Finalmente, tras exponer y comparar los resultados, se presenta el modelo seleccionado, el cual fue implementado en la interfaz para la realización de predicciones.

4.1. VALIDACIÓN

4.1.1. Validación usando Leave One Out.

En primer lugar, se utilizó la técnica de validación Leave One Out, la cual es la más costosa a nivel computacional, ya que requiere realizar tantos entrenamientos como registros existan en el conjunto de datos. Debido a esta característica, el algoritmo SVM no se incluyó en esta técnica de validación, ya que durante el proceso se evidenció un tiempo de ejecución excesivamente alto que impedía completar la tarea. En particular, se observó que cada iteración del SVM tardaba aproximadamente 3 minutos. Considerando que el conjunto de datos cuenta con alrededor de 600 registros, se estimó un tiempo total cercano a 30 horas para completar la validación con esta técnica. Por esta razón, se decidió excluir este modelo del esquema Leave One Out.

Es importante mencionar que, durante el proceso de prueba y exploración, se evidenció una leve mejoría en los resultados al incluir la creatinina en el proceso de entrenamiento, teniendo en cuenta lo importante que es obtener el rendimiento más alto posible, se optó por revertir la decisión de excluir a la creatinina del conjunto de datos, con la finalidad de mejorar el desempeño al máximo.

En Tabla 1 se presentan los resultados obtenidos para cada una de las metricas y cada uno de los modelos usando la validación Leave One Out excluyendo la característica creatinina, más adelante en Tabla 2, se observan los resultados incluyendo la creatinina en el proceso de entrenamiento y prueba, ahí se puede observar que se logró una leve mejoría, este fue el motivo por el cual finalmente se decidió incluir la creatinina.

Algoritmo	Exactitud	Sensibilidad	Especificidad	Precisión	F1 Score	AUC-ROC
KNN	0.859	0.683	0.949	0.872	0.766	0.867
Random Forest	0.878	0.779	0.928	0.847	0.812	0.946

Regresión logística	0.864	0.844	0.875	0.774	0.808	0.9308
XGBoost	0.885	0.774	0.941	0.870	0.819	0.932

Tabla 1 Resultados sin creatinina usando Leave One Out

Posteriormente se obtuvieron los resultados al realizar el proceso de entrenamiento y prueba incluyendo la creatinina en el conjunto de datos. Primero, se usó el algoritmo KNN, en la matriz de confusión (Figura 20) se observan las predicciones hechas por el modelo para cada una de las clases y cuantas de ellas fueron correctas y erróneas.

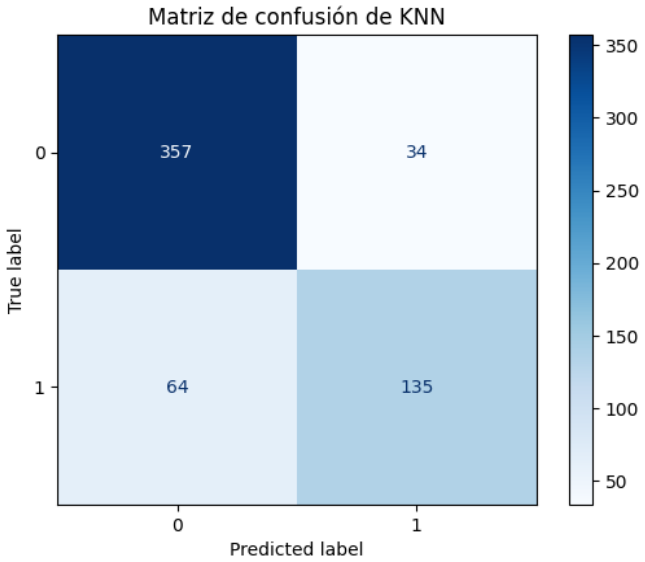

Figura 20 Matriz de confusión de KNN

A partir de estas clasificaciones, se obtuvo que el modelo KNN alcanzó una exactitud del 83,4%, una sensibilidad del 67,8%, una especificidad del 91,3%, una precisión del 79,9% y un F1-score del 73,4%. Si bien la exactitud es moderadamente alta, se observa que la sensibilidad es considerablemente menor, lo que indica que el modelo presenta dificultades para identificar correctamente los casos positivos (diagnosis = 1), es decir, aquellos con presencia de cáncer. Este aspecto resulta especialmente relevante, ya que esta es la clase de mayor interés dentro del problema, y un bajo valor de sensibilidad implica un mayor número de falsos negativos, lo cual es crítico en un contexto médico.

El segundo algoritmo evaluado fue Random Forest. En la Figura 21 se presenta la matriz de confusión generada a partir de sus predicciones. Este modelo obtuvo una exactitud del 89,7%, una sensibilidad del 81,9%, una especificidad del 93,6%, una precisión del 86,7% y un F1-score del 84,2%. En términos generales, este algoritmo mostró una mejora significativa respecto al modelo anterior. No solo alcanzó una mayor exactitud, sino que también presentó un incremento relevante en la sensibilidad, lo que indica una mejor capacidad para identificar correctamente los casos positivos (diagnosis = 1). Además, esta mejora no comprometió la

especificidad, la cual se mantuvo alta, reflejando un buen desempeño en la clasificación de los casos negativos (diagnosis = 0).

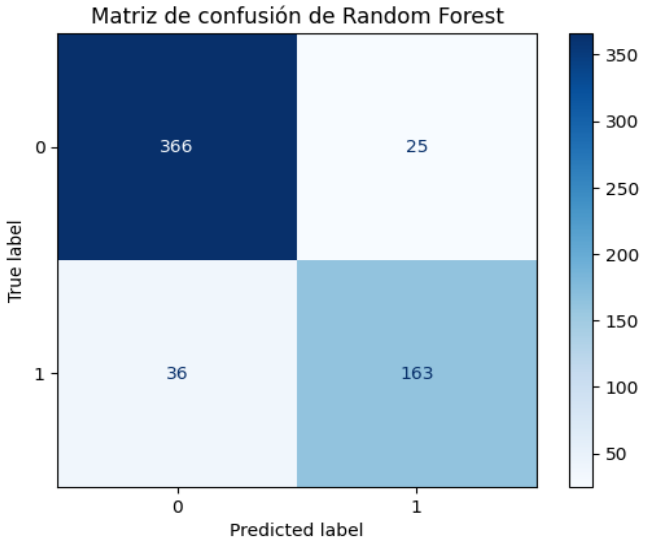

Figura 21 Matriz de confusión de Random forest.

El siguiente modelo evaluado fue la regresión logística. A partir de la matriz de confusión (Figura 22), se calcularon las métricas de evaluación, obteniendo una exactitud del 87,1%, una sensibilidad del 84,9%, una especificidad del 88,2%, una precisión del 78,6% y un F1-score del 81,6%. Aunque este modelo presentó una exactitud inferior a la obtenida por Random Forest, alcanzó la mayor sensibilidad entre los modelos evaluados, lo que indica una mejor capacidad para identificar los casos positivos. Sin embargo, esta mejora se dio a costa de una disminución en la especificidad, lo que indica un mayor número de errores en la clasificación de los casos negativos. En consecuencia, el incremento en la detección de casos positivos comprometió el

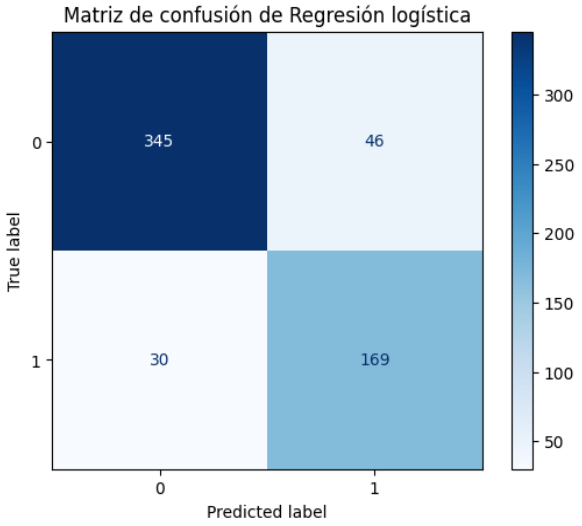

Figura 22 Matriz de confusión de regresión logística.

desempeño en la identificación de pacientes sin cáncer, lo cual explica la reducción en la exactitud del modelo.

Por último, se realizó la evaluación del algoritmo XGBoost, a partir de las clasificaciones se construyó la matriz de confusión (Figura 23), y posteriormente se calcularon las metricas de evaluación, para las cuales se obtuvo un 88.6% de exactitud, una sensibilidad del 77.9%, una especificidad del 94.1%, una precisión del 87.1% y un F1-Score del 82.2%. Si bien este algoritmo alcanzó la especificidad más alta entre todos los algoritmos evaluados, disminuyó considerablemente la sensibilidad, demostrando una baja tasa de acierto en la clasificación de casos positivos (con presencia de cáncer), comprometida por la mejora en la predicción de casos negativos (ausencia de cáncer).

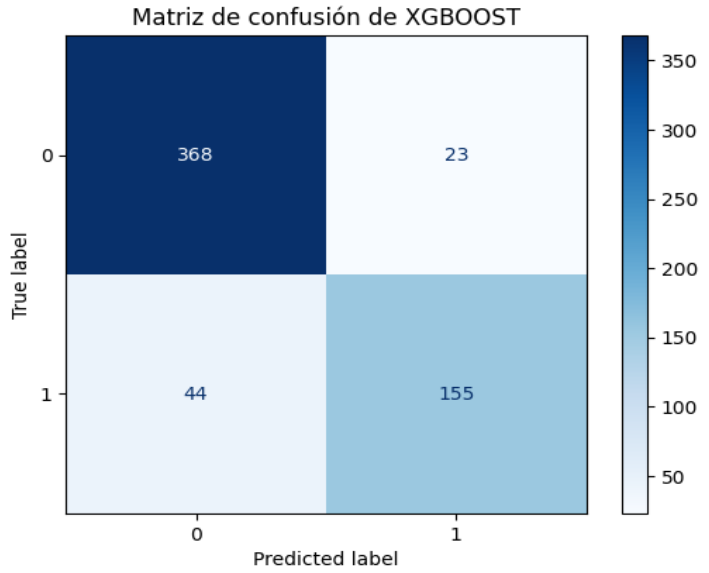

Figura 23 Matriz de confusión de XGBoost

Por último, en Tabla 2 se observa un resumen de las metricas obtenidas por cada uno de los modelos (a excepción de SVM) usando Leave One Out. En ella se puede observar, como ya se mencionó, una leve mejoría respecto a los resultados obtenidos con el conjunto de datos en el que se excluía a la creatinina, esto sustenta la decisión de incluirla. En general se obtuvieron buenos desempeños, resaltando, sobre todo, los obtenidos por el algoritmo Random forest y regresión lineal.

Algoritmo	**Exactitud**	**Sensibilidad**	**Especificidad**	**Precisión**	**F1-Score**	**AUC-ROC**
KNN	0.834	0.678	0.913	0.799	0.734	0.87
Random Forest	0.897	0.819	0.936	0.867	0.842	0.945
Regresión logística	0.871	0.849	0.882	0.786	0.816	0.929
XGBoost	0.886	0.779	0.941	0.871	0.822	0.933

Tabla 2 Resultados obtenidos en Leave One Out.

Por otro lado, se realizó la también la evaluación del conjunto de datos estandarizado con la función MaxAbsScaler, usando esta misma técnica de validación, los resultados obtenidos fueron los siguientes.

Primero se hizo la evaluación del algoritmo Regresión logística y se construyó la matriz de confusión (Figura 24) a partir de sus resultados.

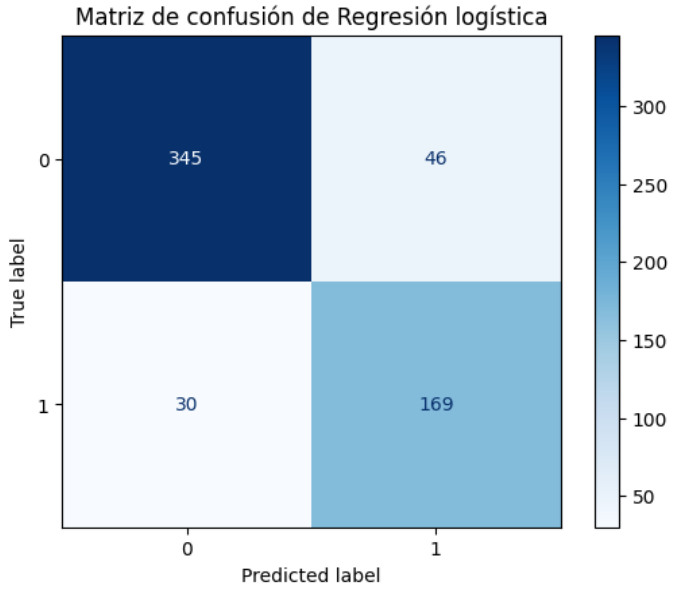

Figura 24 Matriz de confusión LOO con MaxAbsScaler de Regresión Logística.

Posteriormente se repitió el proceso de evaluación con XGBoost, para el cual se obtuvo su respectiva matriz de confusión (Figura 25).

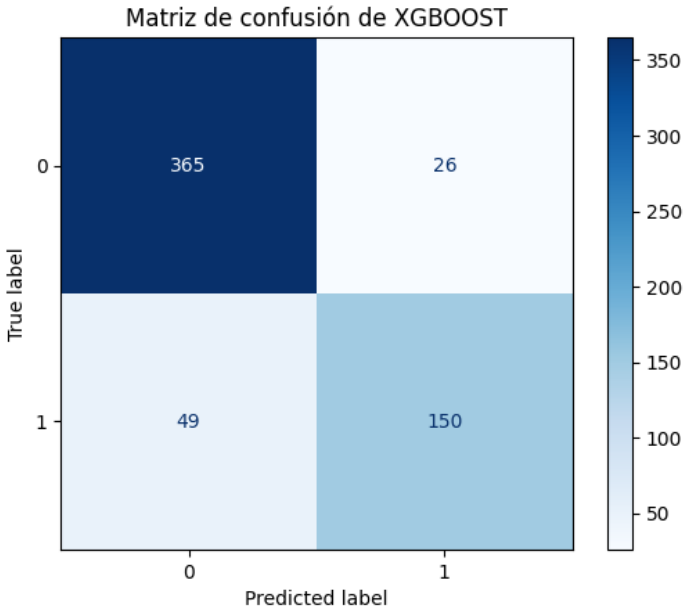

Figura 25 Matriz de confusión LOO con MaxAbsScaler de XGBoost.

El tercer algoritmo evaluado con el conjunto de datos estandarizado con esta función fue Random Forest, para el cual resultó la matriz de confusión observada en la Figura 26.

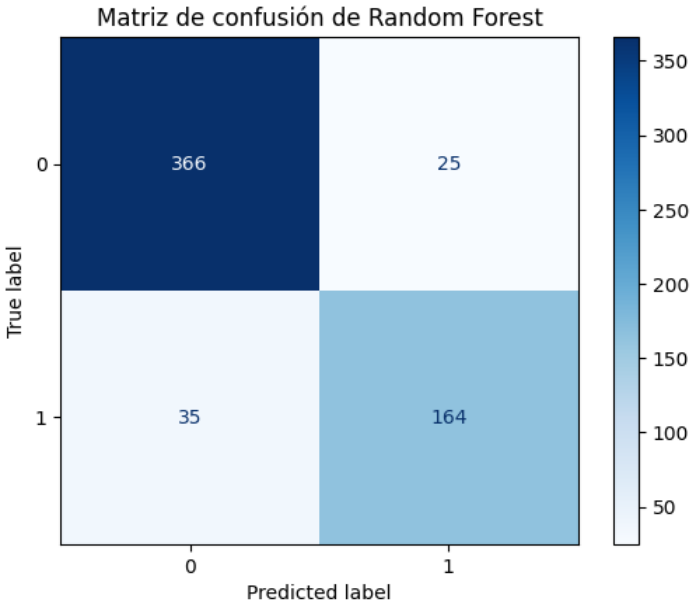

Figura 26 Matriz de confusión LOO con MaxAbsScaler de Random Forest.

Por último, se realizó el entrenamiento y prueba del algoritmo KNN usando este conjunto de datos, de acuerdo con las predicciones realizadas por el modelo se construyó la matriz de confusión (Figura 27).

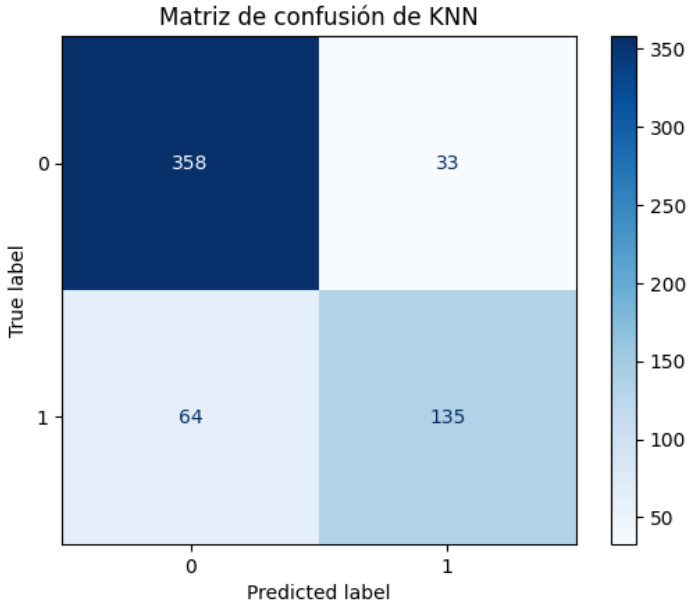

Figura 27 Matriz de confusión LOO con MaxAbsScaler de KNN.

A partir de los resultados de las clasificaciones realizadas por cada uno de los algoritmos se calcularon las metricas de evaluación correspondientes descritas en la .

Algoritmo	Exactitud	Sensibilidad	Especificidad	Precisión	F1-Score	AUC-ROC
XGBoost	0.873	0.754	0.934	0.852	0.8	0.9413
Regresión logística	0.871	0.849	0.882	0.786	0.816	0.929
Random forest	0.898	0.824	0.936	0.868	0.845	0.945
KNN	0.836	0.678	0.916	0.804	0.736	0.864

Tabla 3 Resultados de LOO con MaxAbsScaler.

En la tabla que resume las métricas calculadas para cada algoritmo utilizando la técnica Leave One Out y el conjunto de datos normalizado con MaxAbsScaler (Tabla 3), se observa que no hubo cambios significativos ni mejoras considerables en el desempeño de los modelos. En el caso de la regresión logística, los resultados se mantuvieron iguales, mientras que en Random Forest se aprecia una ligera mejora. Sin embargo, al analizar las matrices de confusión, se evidenció que dicha diferencia correspondía únicamente a una clasificación correcta adicional, por lo que el cambio no resulta lo suficientemente relevante como para concluir que esta técnica de normalización aporta una mejora real al rendimiento de los modelos. De modo que se tomó la decisión de descartar esta función de estandarización para el entrenamiento y prueba de los modelos en las técnicas de validación restantes.

Para finalizar con esta técnica de validación cruzada, se realizó la fase de entrenamiento y prueba con el conjunto de datos estandarizado con la función RobustScaler, para el cual se obtuvieron los resultados mencionados a continuación.

El primer algoritmo evaluado fue KNN, para el cual se obtuvo la siguiente matriz de confusión.

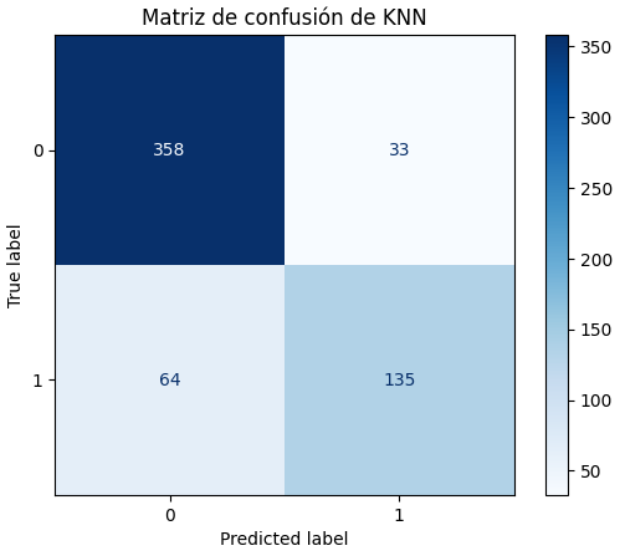

Figura 28 Matriz de confusión con LOO y RobustScaler de KNN.

El segundo algoritmo evaluado con este conjunto de datos fue Regresión logística, para el cual se construyó la matriz de confusión observada en la Figura 29 con respecto a las predicciones hechas por él.

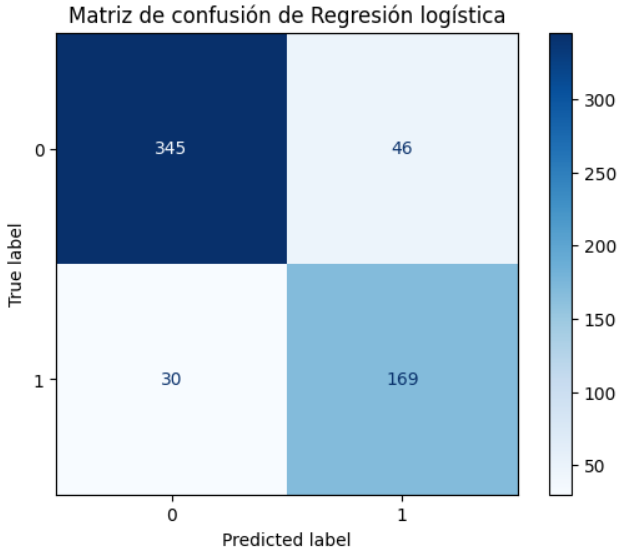

Figura 29 Matriz de confusión LOO con RobustScaler de Regresión logística.

El siguiente algoritmo para el cual se construyó la matriz de confusión (Figura 30) fue XGBoost.

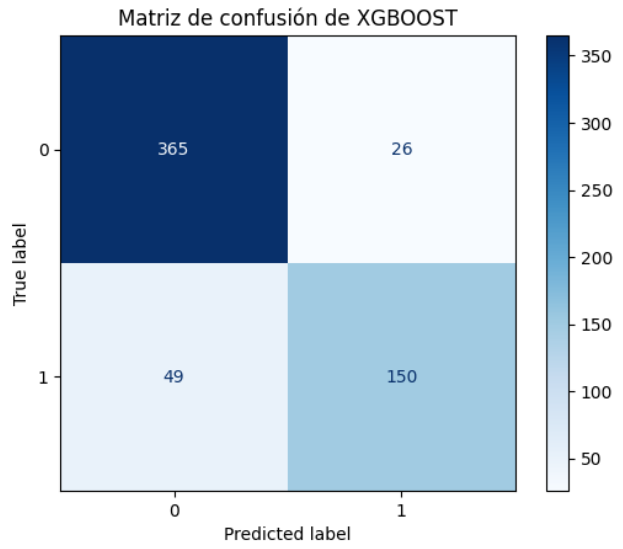

Figura 30 Matriz de confusión LOO con RobustScaler de XGBoost.

Por último, se realizó la evaluación del algoritmo Random Forest, y se obtuvieron las siguientes predicciones.

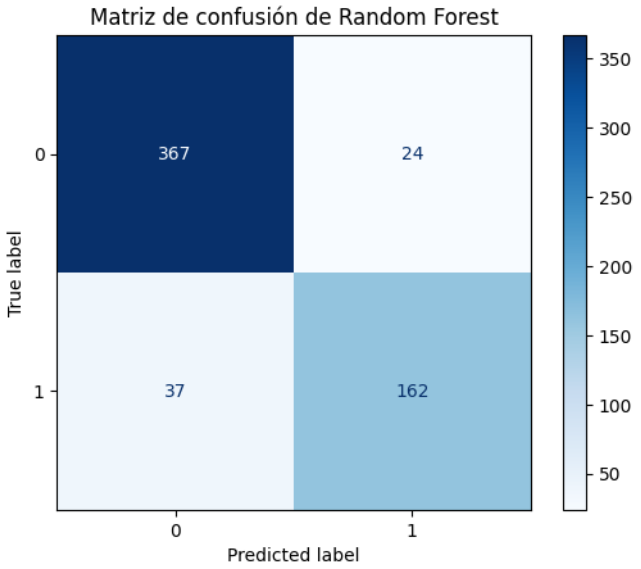

Figura 31 Matriz de confusión LOO con RobustScaler de Random forest.

	Exactitud	Sensibilidad	Especificidad	Precisión	F1-Score	AUC-ROC
KNN	0.836	0.678	0.916	0.804	0.736	0.864
Regresión logística	0.871	0.849	0.882	0.786	0.816	0.929
XGBoost	0.873	0.754	0.934	0.852	0.8	0.941
Random forest	0.897	0.814	0.939	0.871	0.842	0.944

Tabla 4 Resultados de LOO con RobustScaler.

En la tabla anterior (Tabla 4) se presentan los resultados obtenidos para cada métrica utilizando la técnica de validación Leave One Out sobre el conjunto de datos estandarizado con RobustScaler. Al igual que en el caso anterior, no se observan diferencias significativas que permitan evidenciar una mejora considerable en el desempeño general de los modelos. En particular, los resultados de la regresión logística permanecieron iguales, mientras que Random Forest mantuvo la misma exactitud y solo presentó pequeñas variaciones en algunas métricas. Por su parte, KNN mostró una ligera mejora en la exactitud; sin embargo, esta diferencia no fue lo suficientemente relevante como para concluir que la normalización mediante RobustScaler aporta beneficios significativos al rendimiento de los modelos. En consecuencia, también se decidió descartar el uso de esta técnica de normalización en las siguientes fases del experimento. Por lo tanto, las dos técnicas de validación restantes se realizaron únicamente utilizando el conjunto de datos sin aplicar procesos de estandarización.

4.1.2. Validación usando K – FOLD.

Posteriormente, se empleó la técnica de validación K-Fold, utilizando un valor de $K = 10$. En cada iteración se calcularon las distintas métricas de evaluación y, una vez finalizadas las 10 particiones, se obtuvo el promedio de cada una de ellas. Además, debido a que el costo computacional de esta técnica es mucho menor en comparación con Leave One Out, fue posible incluir el algoritmo SVM dentro del proceso de validación. De esta manera, se logró evaluar los cinco modelos seleccionados bajo las mismas condiciones. Por otra parte, además de las métricas tradicionales, se utilizó el área bajo la curva ROC (AUC-ROC) como apoyo en el proceso de selección del modelo con mejor rendimiento. Para ello, se generó la curva ROC correspondiente a cada algoritmo, permitiendo analizar de manera más completa su capacidad para distinguir entre las clases positivas y negativas.

El primer modelo evaluado mediante esta técnica de validación fue la regresión logística. En la Tabla 5 se presentan los promedios obtenidos para cada métrica junto con su respectiva desviación estándar, la cual permite medir qué tan dispersos se encuentran los resultados respecto a la media. En la tabla se observan desviaciones estándar muy pequeñas, lo que indica una baja variabilidad en las métricas entre cada iteración, y, por lo tanto, una gran estabilidad. Además, se graficó la curva AUC-ROC (Figura 32), obteniendo un área bajo la curva igual a 0,927. Este valor, al ser cercano a 1, resulta bastante positivo, ya que indica que el modelo posee una alta capacidad para diferenciar correctamente entre pacientes con y sin cáncer. En consecuencia, los resultados sugieren que la regresión logística presenta un desempeño estable y una buena capacidad discriminativa para el problema planteado.

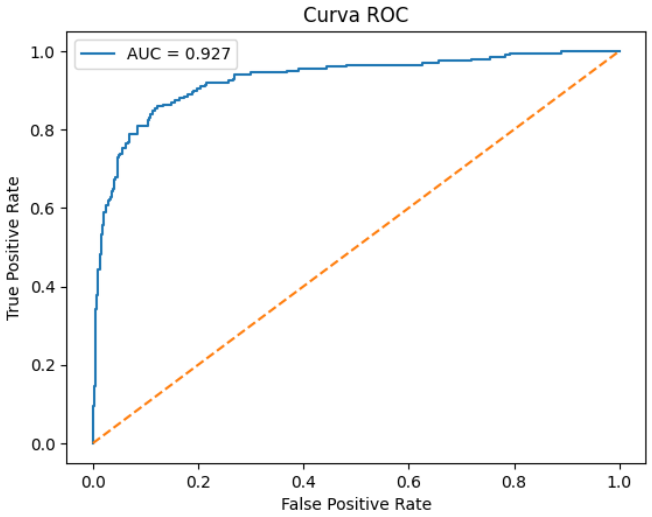

Figura 32 AUC-ROC regresión logística.

El siguiente modelo fue KNN, este modelo además de que obtuvo algunos resultados un poco más bajos, sobre todo en la clasificación de casos positivos, resultó con una mayor variabilidad en algunas de las metricas, alcanzando desviaciones estándar mayores a 0.1 en algunas de ellas, mostrando que en cada iteración el resultado tendía a ser poco estable. Por otra parte, el área bajo la curva de este (Figura 33) fue menor a la obtenida por la regresión logística indicando una menor capacidad discriminativa entre pacientes con y sin cáncer. En resumen,

estos resultados sugieren que KNN presentó un rendimiento, en general, inferior con respecto al modelo anterior.

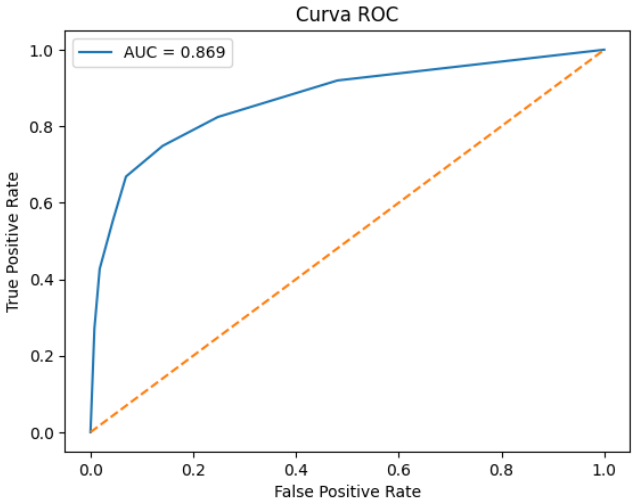

Figura 33 AUC-ROC KNN

En tercer lugar, se utilizó la técnica K-Fold para validar el algoritmo XGBoost. Este modelo obtuvo resultados bastante positivos, alcanzando una alta exactitud y una baja variabilidad en la mayoría de las métricas, lo que demuestra no solo un buen rendimiento, sino también una alta estabilidad entre las diferentes iteraciones. La desviación estándar más alta registrada fue de 0,094 y se presentó en la métrica de sensibilidad, la cual también fue la más baja del modelo. Esto podría indicar ciertas dificultades para identificar correctamente los casos positivos. Sin embargo, a pesar de esta limitación, el desempeño general del algoritmo continuó siendo favorable. Además, al analizar el área bajo la curva ROC (Figura 34), se obtuvo un valor de 0,937, el cual es bastante alto y muestra una muy buena capacidad discriminativa del modelo para diferenciar entre pacientes con y sin cáncer.

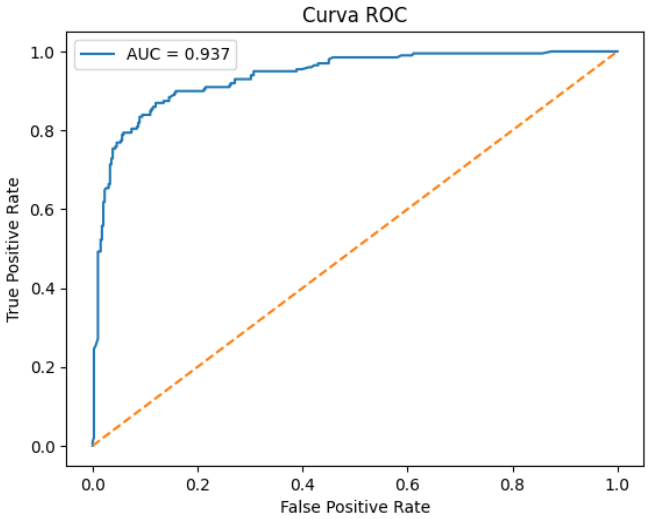

Figura 34 AUC-ROC de XGBoost.

Después se realizó la validación para el algoritmo Random forest. Se puede observar (Tabla 5) que este modelo obtuvo resultados mayores a 0.82 en todas las metricas, mostrando un buen desempeño en la clasificación de ambas clases, además de una baja variabilidad, por lo que, además de un buen rendimiento, el modelo fue bastante estable en cada iteración de la validación. Por otro lado, se obtuvo el resultado más alto de AUC-ROC (Figura 35) entre todos los modelos evaluados, alcanzando un valor de 0.942. En resumen, Random forest obtuvo un rendimiento bastante positivo, con una alta capacidad para clasificar tanto casos positivos como negativos.

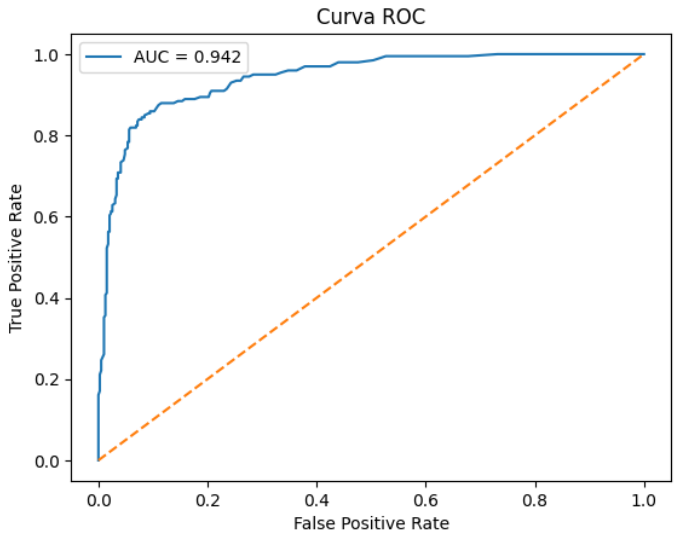

Figura 35 AUC-ROC de Random forest.

Por último, se validó el algoritmo SVM. En términos generales, este modelo obtuvo métricas altas y equilibradas entre sí, lo que evidencia un buen rendimiento. Sin embargo, sus resultados fueron ligeramente inferiores a los obtenidos por el modelo anterior, con excepción de la sensibilidad, en la cual presentó un mejor desempeño. Además, las desviaciones estándar observadas en la tabla fueron bajas, lo que indica una variabilidad reducida entre iteraciones y, por lo tanto, una alta estabilidad durante el proceso de validación. En cuanto al área bajo la curva ROC (Figura 36), el algoritmo alcanzó un valor de 0,922, siendo este bastante alto y con una buena capacidad discriminativa. A pesar de ello, este resultado continuó siendo ligeramente inferior al obtenido por algunos de los modelos evaluados previamente.

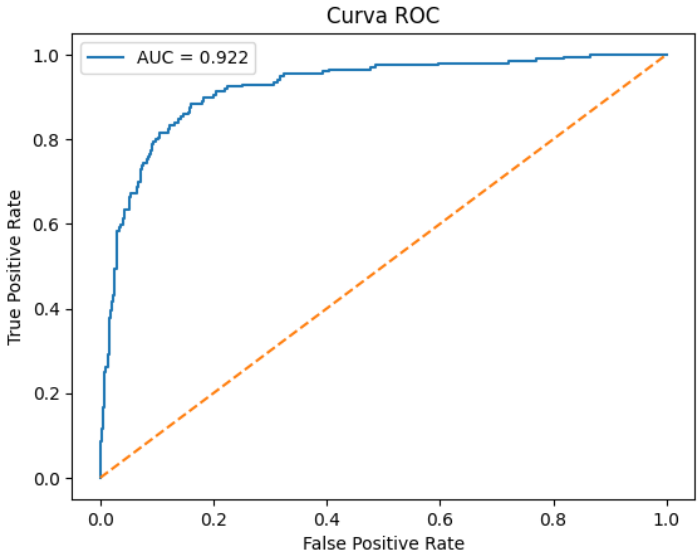

Figura 36 AUC-ROC de SVM.

	Exactitud	Sensibilidad	Especificidad	Precisión	F1-Score
Regresión logística	0.874 ± 0.026	0.852 ± 0.081	0.886 ± 0.041	0.797 ± 0.052	0.819 ± 0.031
KNN	0.842 ± 0.052	0.666 ± 0.133	0.931 ± 0.037	0.825 ± 0.104	0.731 ± 0.113
XGBoost	0.886 ± 0.046	0.773 ± 0.094	0.943 ± 0.034	0.876 ± 0.073	0.819 ± 0.074
Random Forest	0.894 ± 0.052	0.821 ± 0.093	0.933 ± 0.042	0.864 ± 0.082	0.84 ± 0.078
SVM	0.85 ± 0.04	0.871 ± 0.084	0.84 ± 0.04	0.735 ± 0.052	0.795 ± 0.054

Tabla 5 Resultados usando KFOLD.

En la tabla anterior (Tabla 5), como ya se mencionó, se presentan el promedio y la desviación estándar de cada métrica obtenida por los distintos algoritmos evaluados. A partir de estos resultados, se puede observar que los modelos de regresión logística y Random Forest fueron los que alcanzaron el mejor desempeño general, manteniendo además un equilibrio entre todas las métricas y una baja variabilidad entre iteraciones. Tanto en la validación realizada mediante Leave One Out como en la técnica K-Fold, ambos modelos obtuvieron una alta exactitud y una sensibilidad alta, lo cual evidencia una buena capacidad para diferenciar correctamente los casos positivos siendo estos sumamente importantes. Asimismo, las bajas desviaciones estándar muestran que el comportamiento de estos modelos fue consistente y estable a lo largo de las iteraciones.

Por último, antes de seleccionar el modelo que sería implementado en la interfaz, se aplicó una técnica de validación adicional: Hold-Out. La decisión de utilizar este método surgió debido a que, por el alto costo computacional de la validación Leave One Out, no fue posible evaluar el algoritmo SVM con dicha técnica. En consecuencia, con el fin de realizar una

comparación más completa y tomar una decisión mejor fundamentada, se optó por usar una técnica de validación adicional que permitiera analizar el comportamiento de todos los modelos bajo las mismas condiciones. De esta manera, fue posible contar con más evidencia sobre el rendimiento y la estabilidad de cada algoritmo antes de seleccionar el modelo final que sería integrado en la interfaz de usuario.

4.1.3. Validación usando Hold-out.

Para esta técnica se usó una división del conjunto de datos del 75% para entrenamiento y el restante 25% para prueba, además de que se realizaron diez iteraciones, en cada una de ellas se calcularon las métricas y al finalizar las iteraciones se calculó el promedio y la desviación estándar, similar al proceso seguido en la técnica de validación anterior.

	Exactitud	Sensibilidad	Especificidad	Precisión	F1-Score
Regresión logística	0.845 ± 0.028	0.8 ± 0.069	0.869 ± 0.039	0.76 ± 0.05	0.777 ± 0.042
KNN	0.833 ± 0.02	0.656 ± 0.069	0.924 ± 0.026	0.819 ± 0.047	0.725 ± 0.04
XGBoost	0.881 ± 0.021	0.77 ± 0.066	0.938 ± 0.017	0.866 ± 0.031	0.813 ± 0.039
Random Forest	0.886 ± 0.032	0.78 ± 0.081	0.94 ± 0.021	0.87 ± 0.043	0.821 ± 0.056
SVM	0.856 ± 0.023	0.86 ± 0.035	0.855 ± 0.035	0.754 ± 0.044	0.802 ± 0.028

Tabla 6 Resultados usando Hold-out

En resumen, en la tabla anterior se observa que Random Forest, el cual ya había mostrado buenos resultados en las técnicas de validación previas, volvió a obtener un desempeño bastante positivo. Aunque presentó una disminución considerable en la sensibilidad, las demás métricas continuaron siendo muy buenas. En particular, mantuvo la exactitud más alta entre todos los modelos evaluados, así como valores superiores en especificidad, precisión y F1-score. De manera similar, la regresión logística, que previamente había demostrado un rendimiento sólido y equilibrado, conservó métricas altas. Por otro lado, el algoritmo SVM mostró una gran estabilidad en sus resultados, ya que sus métricas variaron muy poco con respecto a las obtenidas en la validación K-Fold. Además, continuó presentando la sensibilidad más alta entre todos, lo que refleja una buena capacidad para identificar los casos positivos. En términos generales, los algoritmos que habían obtenido mejores resultados en las validaciones anteriores mantuvieron un desempeño favorable también en la técnica Hold-Out, lo que aporta mayor confiabilidad al proceso de selección del modelo final.

En esta técnica se redujo significativamente la proporción de datos de entrenamiento, por eso algunos algoritmos con rendimientos buenos y estables en las validaciones anteriores pueden haber experimentado una disminución del desempeño, sin embargo, la información disponible fue suficiente para decidir cual algoritmo se debía implementar en la interfaz de usuario.

4.2. Elección del algoritmo de aprendizaje automático.

Finalmente, a partir de las validaciones realizadas y del análisis de los resultados obtenidos en cada una de las métricas, se decidió implementar en la interfaz el algoritmo Random Forest. En primer lugar, este modelo fue el que alcanzó la mayor exactitud en todas las técnicas de validación aplicadas, lo que indica que logró la mayor cantidad de clasificaciones correctas entre los algoritmos evaluados. Además, aunque la regresión logística presentó una sensibilidad ligeramente superior logrando la sensibilidad más alta, mostrando mayor capacidad para clasificar casos positivos, además de un desempeño bastante equilibrado en todas las métricas, Random Forest consiguió superarla en la mayoría de ellas, situación similar a la observada con el algoritmo SVM. Por otro lado, el valor de AUC-ROC obtenido por Random Forest también fue el más alto entre todos los modelos y, al ser muy cercano a 1, evidencia no solo un mejor rendimiento que los demás, sino también una excelente capacidad discriminativa en general. Este alto valor de AUC-ROC demuestra que el modelo posee una gran capacidad para diferenciar correctamente entre pacientes con y sin cáncer. Asimismo, Random Forest mantuvo valores elevados en todas las métricas evaluadas, lo que indica que logró clasificar adecuadamente tanto los casos positivos como los negativos, a pesar del desbalance presente en el conjunto de datos. Por estas razones, se consideró que este algoritmo era la alternativa más adecuada para ser integrada en la interfaz y utilizada en el proceso de predicción.

4.3. INTERFAZ DE USUARIO

El siguiente paso consistió en desarrollar la interfaz de usuario a partir de los diagramas de actividades y del mockup previamente diseñado. Este proceso facilitó significativamente la implementación, ya que se contaba con una planificación clara del flujo de trabajo que debía seguir la interfaz, así como con una referencia visual del resultado esperado. En general, estos elementos permitieron abordar el desarrollo de manera más estructurada.

Para el desarrollo de la interfaz se utilizó TypeScript como lenguaje de programación, y React como librería fundamental para facilitar el proceso de desarrollo.

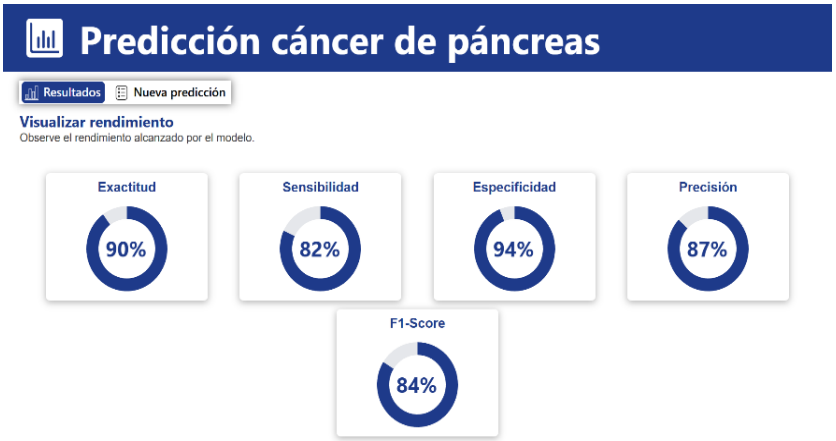

Figura 37 Interfaz: página resultados.

La primera sección desarrollada fue la encargada de mostrar los resultados de las métricas del modelo seleccionado por presentar el mejor rendimiento (Figura 37). En esta sección, las métricas se visualizan en forma de porcentajes y están acompañadas por gráficos de tipo dona, en los cuales la porción azul representa el valor correspondiente de cada métrica. Para la implementación de estas visualizaciones se utilizó la librería “react-chartjs”, lo que permitió presentar la información de manera clara y comprensible para el usuario.

Posteriormente se desarrolló la sección que contiene el formulario que le permitirá al usuario obtener un resultado al diligenciarlo con las características correspondientes (Figura 38). El formulario contiene entradas para cada una de las características escogidas y con las que se realizó el entrenamiento de los modelos. En la página se observan dos contenedores o bloques, uno de ellos contiene el formulario y el otro es el que mostrará los resultados al realizar el envío de la información.

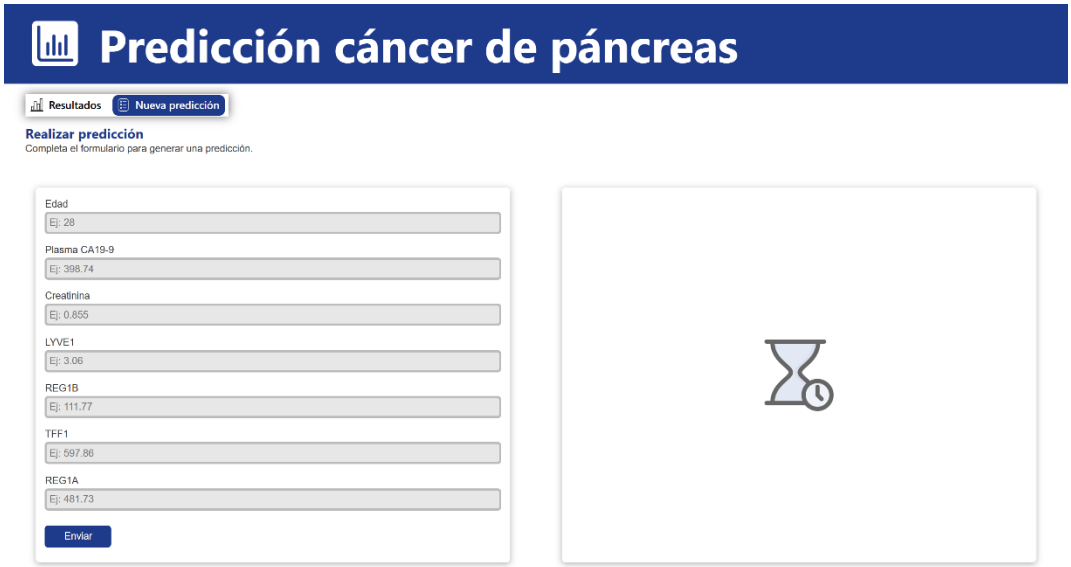

Figura 38 Interfaz: página formulario.

Al enviar la información, antes de mostrar los resultados, se presenta una advertencia (Figura 39) que informa al usuario que la predicción no es definitiva y que no reemplaza un diagnóstico médico. Como se mencionó previamente, esta medida busca evitar interpretaciones erróneas, de modo que el usuario no asuma que el resultado es completamente preciso y comprenda la importancia de acudir a un profesional de la salud en caso de ser necesario.

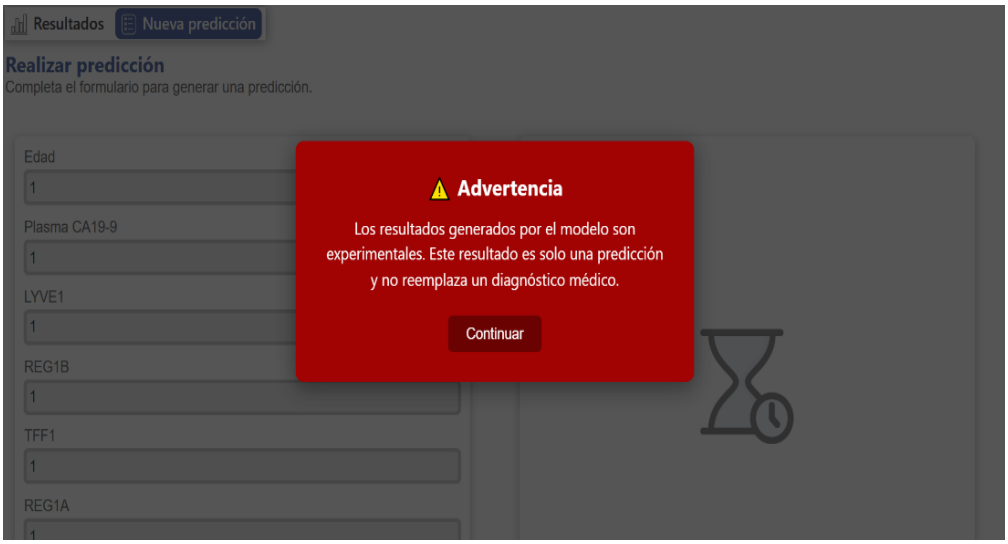

Figura 39 Interfaz: advertencia.

Por último, se desarrolló la presentación de los dos resultados posibles, presencia (Figura 40) y ausencia (Figura 41) de cáncer. Este resultado se muestra al realizar el envío de la información, para ello se deben diligenciar al menos 5 características en total, además de que la edad es obligatoria, por lo que en realidad es obligatorio completar la edad y cuatro características más.

Figura 40 Interfaz: resultado positivo.

Figura 41 Interfaz: resultado negativo.

En la presentación de los resultados se evitó el uso de términos que pudieran interpretarse como concluyentes, como “tienes” o “posees”, con el fin de prevenir malentendidos como los mencionados anteriormente. De esta manera, se busca que el usuario no asuma que las predicciones del modelo son completamente precisas. Aunque el modelo pueda presentar un buen desempeño, siempre existe la posibilidad de error, especialmente en el ámbito médico, donde las condiciones del cuerpo humano pueden variar significativamente entre individuos y no todo es absoluto.

5. CONCLUSIONES

Este proyecto presenta una evaluación de diversos algoritmos de aprendizaje automático para predecir el cáncer de páncreas basado en biomarcadores aprovechando el conjunto de datos público. Para una evaluación completa se consideró el preprocesamiento, el análisis exploratorio de los datos, la selección de características y tres técnicas de validación cruzada. Se observó que el algoritmo de SVM alcanzó la mayor sensibilidad (0.865) en dos técnicas de validación diferentes, seguida de la regresión logística (0.849), lo que indica que fueron más eficaces en el reconocimiento de casos de cáncer. Por otro lado, Random forest alcanzó el resultado más alto en cada una de las metricas restantes en todas las validaciones hechas, lo cual muestra una mayor capacidad de clasificación en general. Los predictores usados fueron la edad, CA19-9, REG1A, REG1B, LYVE1, creatinina y TFF1, y muestran en conjunto una alta importancia en la detección del cáncer demostrada por la alta exactitud alcanzada. Este documento conecta el conocimiento conceptual aprovechando el acceso a antecedentes de investigación y la exploración práctica a través de algoritmos de aprendizaje automático. El proyecto además de alcanzar una exactitud y capacidad discriminativa bastante alta también contribuye a identificar biomarcadores que pueden ayudar en la detección del cáncer de páncreas.

5.1. TRABAJOS FUTUROS

A pesar de haber obtenido resultados significativamente positivos aún hay muchas oportunidades de mejora. Se espera que en futuros trabajos se usen conjuntos de datos mucho más grandes y diversos, que representen de mejor manera a la población mundial para evitar al mayor nivel posible sesgos demográficos. Sería deseable utilizar conjuntos de datos que no solo contengan biomarcadores, sino también antecedentes y otras características medicas de los pacientes que aporten mucho más en la detección temprana del cáncer de páncreas.

Por otro lado, para obtener resultados que sean mucho más exactos y que presenten un mejor desempeño podría pensarse en utilizar modelos de aprendizaje profundo y máquinas con mejores recursos computacionales que permitan una mejor optimización y búsqueda de hiperparámetros, y que no limiten el entrenamiento y prueba de los modelos.

Por último, el paso más importante, este tipo de modelos debería integrarse en sistemas médicos que contengan registros de pacientes, con el propósito de automatizar la detección de cáncer de páncreas por medio de la evaluación de las características y factores de riesgo, de forma que verdaderamente aporte en la detección temprana.

REFERENCIAS BIBLIOGRÁFICAS

[1] W. Chen, B. Zhou, C. Jeon, F. Xie, Y.-C. Lin, R. Butler, Y. Zhou, T. Luong, E. Lustigova, J. Pisegna y B. Wu, «Machine learning versus regression for prediction of sporadic pancreatic cancer,» *Pancreatology,* vol. 23, nº 4, pp. 396-402, 2023.

[2] J.-X. Hu, C.-F. Zhao, W.-B. Chen, Q.-C. Liu, Q.-W. Li, Y.-Y. Lin y F. Gao, «Pancreatic cancer: A review of epidemiology, trend, and risk factors,» *World journal of gastroenterology,* vol. 27, nº 27, p. 4298–4321, 2021.

[3] A. Pacheco Mejías, «Pancreatic cancer, a challenge to the health system,» *Revista Archivo Médico de Camagüey,* vol. 22, nº 5, pp. 847-876, 2018.

[4] S. Tripathi, A. Tabari, A. Mansur, H. Dabbara, C. Bridge y D. Daye, «From Machine Learning to Patient Outcomes: A Comprehensive Review of AI in Pancreatic Cancer,» *Diagnostics,* vol. 14, nº 2, 2024.

[5] V. Landázuri, C. Martínez, J. Gonzáles y M. Proaño, «Cáncer de páncreas. Diagnóstico y tratamiento quirúrgico,» *Polo del conocimiento,* vol. 8, nº 5, pp. 938-951, 2023.

[6] C. Halbrook, C. Lyssiotis, M. Pasca di Magliano y A. Maitra, «Pancreatic cancer: Advances and challenges,» *Cell,* vol. 186, nº 8, pp. 1729-1754, 2023.

[7] S. Debernardi, H. O'Brien, A. Algahmdi, N. Alats, G. Stewart, M. Plješa-Ercegovac, E. Costello, W. Greenhalf, A. Saad, R. Roberts, A. Ney, S. Pereira, H. Kocher, S. Duffy, O. Blyuss y T. Crnogorac-Jurcevic, «A combination of urinary biomarker panel and PancRISK score for earlier detection of pancreatic cancer: A case–control study,» *PLOS Medicine,* vol. 17, nº 12, 2020.

[8] D. Hoff, J. Loscalzo, A. Fauci, D. Kasper, S. Hauser, D. Longo y L. Jameson, «Cáncer de páncreas,» de *Principios de medicina interna de Harrison*, McGraw-Hill Education, 2022.

[9] S. Elsherif, M. Virarkar, S. Javadi, J. Ibarra-Rovira, E. Tamm y P. Bhosale, «Pancreatitis y PDAC: asociación y diferenciación,» *Abdominal Radiology,* vol. 45, pp. 1324-1337, 2020.

[10] «Instituto Nacional del Cáncer,» NIH, [En línea]. Available: https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/sistema-hepatobiliar. [Último acceso: 25 Abril 2026].

[11] R. Smith, «Pancreatitis,» *Salem Press Encyclopedia of Health,* 2026.

[12] H. Daher, S. Punchayil, A. Elbeltagi, R. Ryan Fernandes, J. Jacob, M. Algazzar y M. Mansour, «Advancements in Pancreatic Cancer Detection: Integrating Biomarkers, Imaging Technologies, and Machine Learning for Early Diagnosis,» *Cureus,* vol. 16, nº 3, 2024.

[13] A. Kashikar, S. Maurya, T. Likhar, K. Mirza, A. Kumar y D. Suresh, «Pancreatic Cancer Diagnosis from CT Scan Images Using Machine Learning Methods,» vol. 7, pp. 1589-1595, 2024.

[14] I. Sarker, «Aprendizaje automático: algoritmos, aplicaciones reales y direcciones de investigación,» *Springer Nature Computer Science,* vol. 2, nº 160, 2021.

[15] J. Ungvarsky, «Biomarker.,» de *Salem Press Encyclopedia of Health.*, 2025.

[16] R. Babasaheb, T. Patil y A. Pandey, «Trends in sensing of creatinine by electrochemical and optical biosensors,» *Applied Surface Science Advances,* vol. 19, 2024.

[17] L. Li, «The Investigation of the Correlation between 4 Urine Biomarkers and Intelligent Diagnosis of PDAC,» *IEEE 2nd International Conference on Electronic Technology,* vol. 2, pp. 593-596, 2022.

[18] A. Guevara-Tirado, «Diferencias en los biomarcadores asociados a REG1B en pacientes con y sin pancreatitis crónica.,» *Revista Del Cuerpo Médico Del Hospital Nacional Almanzor Aguinaga Asenjo,* vol. 18, nº 3, pp. 1-16, 2025.

[19] Q.-W. Wang, D.-J. Bian, Z.-Q. Xie, X. Geng, H.-R. Zheng, B.-W. Li, Y.-G. Xu, W.-B. Zou y Z. Liao, «Serum REG1A protein is a biomarker for diagnosis and progression prediction of pancreatitis,» *BMC Gastroenterol,* vol. 26, nº 3, 2026.

[20] G. Luo, K. Jin, S. Deng, H. Cheng, F. Zhiyao, Y. Gong, Y. Qian, Q. Huang, Q. Ni, C. Liu y X. Yu, «Roles of CA19-9 in pancreatic cancer: Biomarker, predictor and promoter,» *Biochimica et Biophysica Acta (BBA) - Reviews on Cancer,* vol. 1875, nº 2, 2021.

[21] V. Da Poian, B. Theiling, L. Clough, B. McKinney, J. Major, J. Chen y S. Hörst, «Exploratory data analysis (EDA) machine learning approaches for ocean world analog mass spectrometry.,» *Frontiers in Astronomy and Space Sciences,* vol. 10, 2023.

[22] F. Dekking, C. Kraaikamp, H. Lopuhaä y L. Meester, «Exploratory data analysis: graphical summaries,» de *A Modern Introduction to Probability and Statistics*, Londres, Springer, 2005, pp. 207-230.

[23] O. Colliot y G. Varoquaux, «Evaluación de modelos de aprendizaje automático y su valor diagnóstico,» *Machine Learning for Brain Disorders,* vol. 197, pp. 601-630, 2023.

[24] S. Shrivastava, A. Yadav y S. Gubta, «A Machine Learning Approach to Load Forecasting Using Extreme Gradient Boosting with Cross-Validation Technique,» *World Conference on Communication & Computing (WCONF),* vol. 3, pp. 1-7, 2025.

[25] I. Tougui, A. Jilbab y J. El Mhamdi, «Impact of the Choice of Cross-Validation Techniques on the Results of Machine Learning-Based Diagnostic Applications,» *Healthcare Informatics Research,* vol. 3, nº 27, pp. 189-199, 2021.

[26] D. Ruggeri y L. Vidács, «KFold Cross-Validation and Early Stopping in Foodomics Neural Networks: Practices, Pitfalls, and Recommendations,» *IEEE Acces,* vol. 13, pp. 190820-190832, 2025.

[27] S. Yadav y S. Shukla, «Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification,» *IEEE 6th International Conference on Advanced Computing (IACC),* vol. 6, pp. 78-83, 2016.

[28] D. Singh y B. Singh, «Feature wise normalization: An effective way of normalizing data,» *Pattern Recognition,* vol. 122, 2022.

[29] M. A. Peshawa y H. F. Rezhna, «Data Normalization and Standardization: A Technical Report,» *The Machine Learning Lab,* vol. 1, nº 1, pp. 1-6, 2014.

[30] «Scikit-Learn,» [En línea]. Available: https://scikit-learn.org/stable/modules/preprocessing.html#standardization-or-mean-removal-and-variance-scaling. [Último acceso: 25 Abril 2026].

[31] M. Jaramillo, J. Ruano, D. Bravo, S. Medina, M. Gómez, F. Gonzáles y E. Romero, «Automatic Localization of Pancreatic Tumoral Regions in Whole Sequences of Echoendoscopy Procedures,» *International Symposium on Medical Information Processing and Analysis (SIPAIM),* vol. 19, pp. 1-5, 2023.

[32] T. K. Singh, R. Sonavane, P. Tripathi, J. P. Panchal, S. Sharma y S. M. Jaiswal, «A Machine Learning Framework for Liver Cancer Classification Using Clinical Features,» *2026 International Conference on Emerging Smart Computing and Informatics (ESCI),* pp. 1-6, 2026.

[33] G. Klöppel y V. Adsay, «Chronic Pancreatitis and the Differential Diagnosis Versus Pancreatic Cancer,» *Archives of Pathology & Laboratory Medicine.,* vol. 133, nº 3, pp. 382-387, 2009.

[34] T. Alhobaya, R. Peravali y M. Ashkar, «The Relationship between Acute and Chronic Pancreatitis with Pancreatic Adenocarcinoma: Review.,» *Diseases,* vol. 9, nº 4, 2021.

[35] G. Lewison, G. Owen, H. Gomez, E. Cazap, R. Murillo, K. Unger Saldaña, M. Dreyer, A. Tsunoda y J. Jimenez de la Jara, «Cancer Research in Latin America, 2014-2019, and its disease Burden,» *Journal of Scientometric,* vol. 10, nº 1, pp. 21-31, 2021.

[36] M. Busch y N. Dünker, «Trefoil factor family peptides – friends or foes?,» *Biomolecular Concepts,* vol. 6, nº 5, pp. 343-359, 2015.

[37] S. Prest, F. May y B. Westley, «The estrogen-regulated protein, TFF1, stimulates migration of human breast cancer cells,» *The Faseb Journal,* vol. 16, pp. 592-594, 2002.

[38] J. Alqedra, L. Elarnaut, A. Ethman, A. Elnahry, H. A. Serhan y M. Elhadi, «The role of Trefoil Family Factor 1 (TFF1) as a potential marker for retinoblastoma: a systematic review and meta-analysis,» *BMC Ophthalmology,* vol. 26, nº 1, p. 151, 2026.

[39] J. Lyu, M. Jiang, Z. Zhu, H. Wu, H. Kang, X. Hao, S. Cheng, H. Guo, X. Shen, T. Wu, J. Chang y C. Wang, «Identification of biomarkers and potential therapeutic targets for pancreatic

cancer by proteomic analysis in two prospective cohorts,» *Cell Genomics,* vol. 4, nº 6, 2024.

[40] F. Tajik, S. Kamali Zonouzi y S. Razi, «Tumor Markers Involved in Invasion of Pancreatic Cancer,» *Immunol Genet J,* vol. 8, nº 3, pp. 273-287, 2025.

[41] «XGBoost,» [En línea]. Available: https://xgboost.readthedocs.io/en/release_3.2.0/python/python_api.html#xgboost.XGBClassifier. [Último acceso: 3 Mayo 2026].

[42] «miro,» [En línea]. Available: https://miro.com/es/mockup/que-es-mockup/. [Último acceso: 4 Mayo 2026].