UNIVERSIDAD NACIONAL DE LA PLATA

FACULTAD DE INFORMÁTICA

TESINA DE GRADO

Primeras Experiencias en la Identificación de Personas con Riesgo de Diabetes en la Población Argentina usando Técnicas de Aprendizaje Automático

Autor: Gonzalo Tittarelli

Directores: Dr. Franco Ronchetti y Dr. Enzo Rucci

La Plata, Argentina

Septiembre 2023

Agradecimientos

Quiero expresar mi más profundo agradecimiento a todas las personas que, de una u otra manera, contribuyeron a la realización de mi tesina.

A la Facultad de Informática de la UNLP por alojarme estos años y brindarme las herramientas necesarias para desempeñarme como profesional. A sus directivos, docentes y compañeros, quienes compartieron este hermoso y largo camino conmigo.

Mis más sinceros agradecimientos hacia mis directores, Enzo y Franco, cuya disposición y orientación fueron esenciales en el desarrollo de mi tesina.

A mi mamá y mi hermana, por su ayuda y apoyo en este camino.

A mi Cieli, por su contención, ayuda y paciencia: Amorcita, gracias por no dejarme rendir nunca y por enseñarme que, sin importar el camino, siempre se llega.

Quiero dedicar un especial agradecimiento a mi papá, en estas pocas palabras:

Viejo, me enseñaste a luchar ante las adversidades de la vida, inculcándome valores como el respeto y la bondad hacia los demás. A no dejarme bajar los brazos por haber elegido otra carrera. A siempre guiarme por el camino de la humildad y el trabajo. Tu ejemplo me mostró que lo esencial es amar lo que hago, sin importar si está bien o mal hecho. Gracias por haber sido como fuiste: un gran padre y, por sobre todo, un gran amigo.

Resumen

La detección de Diabetes Tipo 2 (DT2) y prediabetes (PDM) representa un verdadero desafío para la medicina debido a la ausencia de síntomas patogenómicos y a la falta de conocimiento de los factores de riesgo asociados. El diagnóstico tardío de esta enfermedad puede desencadenar complicaciones graves que impactan significativamente en la calidad de vida y generan un aumento en los gastos médicos. Asimismo, la remisión de la DT2 es posible en algunas personas, por lo que la detección temprana y el control son cruciales. Si bien existen algunas propuestas de modelos de aprendizaje automático que permiten identificar a personas en riesgo, las características de esta enfermedad hacen que uno que resulte adecuado para una población, no necesariamente lo sea para otra. La presente investigación propone desarrollar y evaluar modelos predictivos que permitan identificar personas con riesgo de DT2 y PDM específicos en el contexto de la población Argentina. Tras llevar a cabo una cuidadosa caracterización y preprocesamiento del conjunto de datos disponible, se generaron diversas segmentaciones del mismo, considerando el equilibrio entre la cantidad de registros y de variables disponibles, además del propósito de predicción. Luego de aplicar diferentes modelos predictivos de clasificación y regresión, los resultados obtenidos muestran que algunos de ellos obtuvieron muy buenos rendimientos para algunas de las segmentaciones planteadas. En particular, los modelos *Random Forest*, Árbol de Decisión y Redes Neuronales Artificiales exhibieron un destacado poder predictivo, al alcanzar valores considerables en las métricas evaluadas. Debido a limitaciones propias de la base de datos, no es posible afirmar que los resultados sean concluyentes, aunque sí resultan promisorios. Dada la carencia de herramientas de este tipo para la población Argentina, este estudio constituye un punto de partida fundamental hacia la creación de modelos más sofisticados.

TABLA DE CONTENIDOS

Página

ÍNDICE DE TABLAS

TABLA **Página**

Índice de figuras

Figura **Página**

Listado de abreviaturas

R^2 Coeficiente de Determinación.

AMGS Automonitorización de la Glucosa en Sangre.
ANN Redes Neuronales Artificiales.
AUC Área Bajo la Curva.

BCE Entropía Cruzada Binaria.

CENEXA Centro de Endocrinología Experimental y Aplicada.
CONICET Consejo Nacional de Investigaciones Científicas y Técnicas.
CRISP-DM Cross Industry Standard Process for Data Mining.
CV Validación Cruzada.

DC Dataset Clínica.
DC-bin Dataset Clínica Binario.
DCG Dataset Clínica+Glucemia basal.
DCG-bin Dataset Clínica+Glucemia basal Binario.
DCL Dataset Clínica+Laboratorio.
DCL-bin Dataset Clínica+Laboratorio Binario.
DL Dataset Laboratorio.
DL-bin Dataset Laboratorio Binario.
DM Diabetes Mellitus.
DT Árbol de Decisión.
DT1 Diabetes Tipo 1.
DT2 Diabetes Tipo 2.

ECV Enfermedades Cardiovasculares.
ENFR Encuesta Nacional de Factores de Riesgo.

FID Federación Internacional de Diabetes.
FINDRISK FINnish Diabetes Risk SCore.
FN Falsos Negativos.
FP Falsos Positivos.
FPR False Positive Rate.
FRCV Factores de Riesgo Cardiovasculares.

GAA Glucemia Alterada en Ayunas.

HDL Lipoproteínas de Alta Densidad.

IA Inteligencia Artificial.
IG Ganancia de Información.
IMC Índice de Masa Corporal.

k-NN K vecinos más cercanos.

LDA Análisis Discriminante Lineal.
LDL Lipoproteínas de Baja Densidad.
LOR Regresión Logística.
LR Regresión Lineal.

MAE Error Absoluto Medio.
MC Matriz de Correlación.
mg/dL Miligramos de Azúcar por Decilitro.
ML Aprendizaje Automático.
MSE Error Cuadrático Medio.

NB Naive Bayes.
NCA Análisis de Componentes de Vecindad.
NGSP National Glycohemoglobin Standardization Program.

NHANES Encuesta Nacional de Examen de Salud y Nutrición.
NLP Procesamiento de Lenguaje Natural.

OMS Organización Mundial de la Salud.

PC Componentes Principales.
PCA Análisis de Componentes Principales.
PDM Prediabetes.
PPDBA Programa Piloto de Prevención Primaria de Diabetes en la provincia de Buenos Aires.
PR Precision-Recall.
PTOG Prueba de tolerancia oral a la glucosa.

ReLu Unidad Lineal Rectificada.
RF Random Forest.
RIC Rango Intercuartil.
RMSE Raíz de Error Cuadrático Medio.
ROC Receiver Operating Characteristic.

SLR Regresión Lineal Simple.
SVM Support Vector Machine.

t-SNE Incrustación de Vecinos Estocásticos Distribuidos en t.
Tanh Tangente Hiperbólica.
TGA Tolerancia a la Glucosa Alterada.
TN Verdaderos Negativos.
TP Verdaderos Positivos.
TPR True Positive Rate.

UNLP Universidad Nacional de La Plata.

CAPÍTULO 1

INTRODUCCIÓN

En una primera instancia, se expone la motivación que impulsó la realización de la presente investigación (Sección 1.1). Posteriormente, se exponen los objetivos perseguidos, se ofrece una descripción detallada de la metodología empleada (Sección 1.2), y se presentan las contribuciones obtenidas (Sección 1.3). Para concluir, se proporciona una visión general de la estructura y organización de este documento (Sección 1.4).

1.1 Motivación

La Diabetes Mellitus (DM), comúnmente conocida como diabetes, es una enfermedad crónica no transmisible caracterizada por la incapacidad del cuerpo para metabolizar el azúcar en sangre (glucosa) de manera eficiente. La regulación de la metabolización de la glucosa depende de una hormona denominada insulina, cuyo órgano productor es el páncreas. El principal objetivo de la insulina es permitir que la glucosa, proveniente de los alimentos que consumimos, ingrese a las células del cuerpo convirtiéndose en energía. La ausencia total o parcial en la producción de insulina, o bien la ineficacia del organismo para utilizarla, conduce a un aumento permanente en los niveles de glucosa en sangre (glucemia) por encima del valor normal [1]. El efecto de este estado no controlado se lo conoce como hiperglucemia y es un indicador clínico de esta enfermedad.

La Diabetes Tipo 1 (DT1), representa alrededor del 10% de las personas que padecen esta enfermedad, y es aquella en la que el páncreas produce poca o nula insulina por sí mismo [2], lo que significa que las personas necesitan inyecciones diarias de esta hormona para vivir (insulinodependiente). La forma más frecuente, representada por casi el 90% de los casos, es la Diabetes Tipo 2 (DT2). En esta variante el cuerpo es capaz de producir insulina, pero se vuelve resistente a ella de a poco, pudiendo llegar a ser insuficiente con el tiempo [3].

En la actualidad, la DM presenta una situación preocupante para la salud mundial de cara a futuro. En 2019 la Federación Internacional de Diabetes (FID) confirmó que la DM era una de las emergencias de salud que crecía de manera vertiginosa alcanzando niveles alarmantes: casi 500 millones de personas a nivel mundial vivían con esta enfermedad. Se estimó que esta cifra seguiría en aumento alcanzando los 700 millones para 2045 [4]. Esta afección representa una gran amenaza para la salud, ignorando el estado socioeconómico y las fronteras entre los países. En ese sentido, Argentina no es la excepción. Según la 4ta Encuesta Nacional de Factores de Riesgo (ENFR), realizada en 2018, la prevalencia de glucemia elevada o diabetes por autorreporte en la población adulta aumentó de 9,8% en 2013 a un 12,7%, en concordancia con el exceso de peso, uno de los principales factores de riesgo de la DM. En tanto, la prevalencia combinada de glucemia elevada o diabetes en mayores de 18 años es de 10,9%. Entre quienes no se autorreportaron como diabéticos ni refirieron jamás haber tenido una glucemia elevada, el 5% tuvo la glucemia elevada y 20% de la población presenta alto o muy alto riesgo de desarrollar DT2 a 10 años [5].

La Tolerancia a la Glucosa Alterada (TGA), la Glucemia Alterada en Ayunas (GAA) y su combinación (TGA+GAA) representan estadios previos a la DM en donde el nivel de glucemia no se encuentra suficientemente alto para diagnosticarla, pero si excedido [6]. La Prediabetes (PDM) es un término frecuentemente en uso que hace alusión a estas afecciones y cada vez toma mayor relevancia, ya que implica el riesgo posterior de desarrollo de DT2 y complicaciones diabéticas. Su evolución hacia DT2 se relaciona con la gravedad del nivel de glucemia y factores de riesgo como la edad y el peso. En 2019, la FID estimó que 33,9 millones de adultos de entre 20 y 79 años padecían PDM en América del Sur y Central esperándose un aumento a 48,1 millones en 2045 [4].

Los números alarmantes de la DM no diagnosticada se deben a que los síntomas de

la DT2 suelen ser leves (o estar ausentes) y al desconocimiento de los factores de riesgo asociados, lo que los convierte en un proceso lento y progresivo. En consecuencia, durante ese tiempo la enfermedad se desarrolla de manera "silenciosa", siendo frecuente que transcurra un largo periodo de tiempo asintomático.

La DM y PDM no diagnosticada a tiempo o mal controlada puede conducir al desarrollo y progresión de complicaciones agudas y graves crónicas que afectan al corazón y vasos sanguíneos (enfermedades cardiovasculares, como la insuficiencia cardiaca, infarto de miocardio y accidentes cerebrovasculares), los ojos (retinopatía diabética), los riñones (nefropatía diabética), los nervios (neuropatía diabética), complicaciones del embarazo y orales (periodontitis) [7], [8], generando un fuerte deterioro en la calidad de vida de las personas (pudiendo ser incapacitantes), aumentando los costos en salud y hasta poniendo en peligro la vida.

Afortunadamente, se ha demostrado que las intervenciones tempranas destinadas a generar un cambio en el estilo de vida retrasan o previenen la DT2 y sus complicaciones [8]–[11]. Se cree que la DM se encuentra interrelacionada fuertemente con factores de comportamiento como la mala alimentación, el sedentarismo [3], [8], aunque también se encuentra condicionada por factores genéticos y ambientales [2]. Debido a esto, cada vez hay más pruebas que demuestran que la remisión de DT2 es posible en algunas personas [4]. Por esta razón, la detección temprana de DM y PDM para su control resulta ser un desafío importante.

El Aprendizaje Automático (en adelante ML, por sus siglas en inglés) es una rama de la Inteligencia Artificial (IA) que se centra en el estudio de mecanismos que brindan a las máquinas la capacidad de aprender, sin ser explícitamente programadas. Los sistemas aprenden a generalizar el conocimiento a partir de las experiencias o datos de ejemplo (aprendizaje inductivo), a diferencia de la IA, dando lugar a diferentes modelos capaces de realizar predicciones o clasificaciones en una amplia variedad de dominios incluso hasta donde el conocimiento humano se encuentra limitado.

Si bien la terminología de ML se remonta a los años 50 [12], el aumento de la capacidad de computación y los grandes volúmenes de datos condujeron a un crecimiento exponencial en los últimos años, teniendo una amplia aplicabilidad en diferentes áreas como la Medicina, Genética, Ingeniería, Robótica, Lingüística, Videojuegos, Web, Big Data, etc. En los últimos años ha habido decenas de estudios que buscan detectar tempranamente diferentes tipos de enfermedades como el cáncer de mama [13], retinopatías que pueden causar ceguera [14], esquizofrenia [15] y Alzheimer [16] entre otros, así como ayudar a los profesionales a la toma de decisiones informadas y eficientes mejorando el diagnóstico y la calidad general de la atención médica.

Particularmente en Argentina existe el Programa Piloto de Prevención Primaria de Diabetes en la provincia de Buenos Aires (PPDBA) [17] desarrollado por el CENEXA (UNLP-CONICET) y financiado por el Ministerio de Ciencia y Tecnología de la Nación, la empresa SANOFI y el CONICET [18]. El programa cuenta con una base de datos la cual podría utilizarse para desarrollar modelos predictivos para la identificación de personas con alto riesgo de padecer DM y PDM en su población. Dicho estudio se realizó en tres municipios de la provincia de Buenos Aires utilizando el cuestionario FINnish Diabetes Risk SCore (FINDRISK) [19] para identificar personas con alto riesgo de desarrollar DT2.

A partir de este cuestionario se derivan ciertas variables clínicas, presentes en la base de datos, como la edad, índice de masa corporal, y la frecuencia de actividad física realizada. Adicionalmente, la base de datos, cuenta con variables de laboratorio asociadas a los Factores de Riesgo Cardiovasculares (FRCV) como el colesterol de Lipoproteínas de Baja Densidad (LDL, por sus siglas en inglés), Lipoproteínas de Alta Densidad (HDL, por sus siglas en inglés) y triglicéridos, las cuales se relacionan con el diagnóstico de DM.

En base a todo lo expuesto, es posible concluir que la diabetes es una enfermedad de difícil detección debido a la ausencia de síntomas específicos y/o falta de conocimiento de los factores de riesgo asociados. En muchas oportunidades, la detección temprana de la misma es muy compleja, viéndose esto reflejado en la alta cantidad de pacientes que padecen esta enfermedad sin conocimiento alguno hasta alcanzar niveles irreversibles. En ese sentido, resulta de suma importancia realizar un estudio y comparación de diferentes modelos predictivos centrados en identificar aquellas personas con alta probabilidad de padecer DM y PDM en la población Argentina. Esta investigación no busca reemplazar los mecanismos de diagnóstico de esta enfermedad, sino ayudar a identificar quienes deban realizarse las pruebas pertinentes con el objetivo de obtener un diagnóstico temprano para su rápido control. De este modo, lograríamos evitar que complicaciones futuras, por desconocimiento, afecten la calidad de vida de las personas.

1.2 Objetivo y metodología

El objetivo general de esta tesina consiste en proponer y evaluar modelos predictivos que permitan una detección efectiva de aquellas personas con riesgo de diabetes en la base de datos del programa PPDBA utilizando diferentes técnicas de ML. Mediante la identificación temprana de estas afecciones, se buscará prevenir y mitigar la progresión y desarrollo de complicaciones agudas y graves asociadas a dicha enfermedad en la población Argentina. Para esto, se definen los siguientes objetivos específicos:

- Explorar las propuestas existentes para modelos predictivos de DM y PDM a fin de realizar su caracterización correspondiente.
- Caracterizar la base de datos del programa PPDBA.
- Diseñar y desarrollar modelos predictivos de DM y PDM empleando técnicas de ML que sean aplicables a la base de datos del programa PPDBA.
- Evaluar y comparar el rendimiento de los modelos desarrollados usando métricas usuales en el área.

Para poder cumplir con los diferentes objetivos se realizaron las siguientes actividades:

- Se estudiaron los fundamentos de los temas de base, como DM y ML.
- Se relevó la bibliografía existente en la temática a partir de la búsqueda en bases de datos especializadas.

- Se estudió, caracterizó y preprocesó la base de datos del programa PPDBA incluyendo un análisis de la posibilidad de existencia de datos nulos y su tratamiento.
- Se diseñaron y desarrollaron modelos utilizando diferentes técnicas de ML partiendo de la base de datos del programa PPDBA, considerando distintos hiperparámetros y variables de entrada. Particularmente, se desarrollaron modelos de Redes Neuronales Artificiales, *Random Forest*, Árbol de Decisión, Regresión Lineal, Regresión Logística y K vecinos más cercanos.
- Se midió el rendimiento de cada modelo desarrollado, empleando métricas usuales del área, como *accuracy*, *precision*, *recall*, *F1-score*, Error Absoluto Medio, Error Cuadrático Medio, Coeficiente de Determinación, el área bajo la curva ROC y *Precision-Recall*.
- Se evaluó y comparó el rendimiento obtenido por los modelos predictivos en búsqueda de la mejor alternativa, considerando diferentes escenarios posibles.

El preprocesamiento de la información, así como la implementación, evaluación y visualización de los modelos se realizó utilizando el lenguaje *Python* y librerías de alto nivel, como *Pandas* [20], [21], *NumPy* [22], *Matplotlib* [23], *TensorFlow* [24], y *scikit-learn* [25].

1.3 Contribuciones

Entre las contribuciones de este trabajo, se pueden mencionar:

- La definición de diferentes segmentaciones (*datasets*) a partir de la base de datos del programa PPDBA, considerando el compromiso entre cantidad de variables y de registros disponibles, además del propósito de predicción.
- Un conjunto de diferentes modelos predictivos que permiten identificar personas con riesgo de padecer DM y PDM en la población Argentina. Tanto los modelos predictivos como las distintas segmentaciones generan un conjunto de posibilidades que, a priori, amplían el espectro para abordar distintas necesidades a futuro, además de enriquecer la presente investigación.
- Un análisis comparativo riguroso de los modelos desarrollados ante diferentes escenarios posibles, el cual sienta las bases para modelos de predicción más sofisticados del riesgo de padecer esta enfermedad en Argentina.

Por último, vale la pena mencionar que parte de los resultados obtenidos en esta tesina han sido incluidos en la siguiente publicación: *"Primeras Experiencias en la Identificación de Personas con Riesgo de Diabetes en la Población Argentina utilizando Técnicas de Aprendizaje Automático"*. Enzo Rucci, Gonzalo Tittarelli, Franco Ronchetti, Jorge Elgart,

Laura Lanzarini, Juan José Gagliardino. Aceptado en: XXIX Congreso Argentino de Ciencias de la Computación (CACIC 2023), a desarrollarse en Octubre de 2023:[1].

1.4 Organización del documento

Finalizando este primer capítulo, se describe la organización del presente documento:

- En el Capítulo 2, se abordan una serie de conceptos fundamentales vinculados a la DM, que abarcan su fisiología general, tipos, causas y consecuencias hasta los posibles tratamientos. Asimismo, se proporcionan varios conceptos relacionados con el ML, incluyendo su definición, tipos de algoritmos, metodología utilizada en sus procesos, y una descripción de los modelos y métricas utilizadas en la presente investigación. Finalmente, se describe el estado del arte respecto a la identificación de personas con riesgo de diabetes a nivel global utilizando técnicas de ML.

- En el Capítulo 3, se expone una descripción y caracterización del conjunto de datos así como el preprocesamiento realizado sobre el mismo. Luego, se describen las segmentaciones planteadas sobre el conjunto de datos disponible, enumerando las ventajas y desventajas de cada una. Finalmente, se establecen aquellas consideraciones a tener en cuenta antes de afrontar las experimentaciones propuestas.

- En el Capítulo 4, se experimentan para cada segmentación propuesta los diferentes modelos de clasificación binaria aplicados, describiendo así los resultados obtenidos. Con el objetivo de identificar aquellos atributos relevantes, se realizará un proceso de selección de *features*. Asimismo, se propone una experimentación y evaluación de diferentes modelos de regresión. Finalmente, se procede con un análisis y comparación general respecto a los resultados obtenidos.

- En el Capítulo 5, se presentan las conclusiones a las que se llegaron respecto a los objetivos propuestos y una serie de posibles trabajos futuros que pueden desarrollarse a partir de la labor realizada en la presente investigación.

[1] https://cacic2023.unlu.edu.ar/congreso/papersAceptados.html

CAPÍTULO 2

MARCO TEÓRICO Y ESTADO DEL ARTE

En el presente capítulo, se abordará el marco teórico referido a los dos temas principales relacionados. Inicialmente, se pondrá foco en la DM (Sección 2.1) para luego profundizar en ML (Sección 2.2). Finalmente, se realizará una reseña del estado de arte (Sección 2.3).

2.1 Diabetes

Como se mencionó anteriormente (Sección 1.1), la DM es una enfermedad crónica no transmisible caracterizada por una presencia elevada del nivel de azúcar en sangre (glucosa) por encima del valor normal, condición conocida como hiperglucemia. A continuación, se realizará una breve descripción fisiológica del proceso de regulación de la glucosa en el organismo, que ayudará a comprender las causas de la DM (2.1.1). Luego se presentarán los tipos (2.1.2), causas (2.1.3), síntomas y diagnóstico (2.1.4), y complicaciones (2.1.5) asociadas con la afección. Finalmente, se describe el impacto sobre la población mundial (2.1.6) culminando esta sección con los tratamientos requeridos por las personas que padecen esta enfermedad (2.1.7).

2.1.1 Fisiología general

A fin de comprender los efectos producidos por esta enfermedad, resulta de suma importancia entender cómo funciona el cuerpo de una persona sin DM. Nuestro organismo necesita adquirir constantemente sustancias del medioambiente para sobrevivir como el oxígeno, agua y alimentos. En particular, los alimentos que ingerimos se terminan descomponiendo en moléculas como los azúcares, los lípidos o las proteínas. Estas son aprovechadas por el organismo para construir o renovar las estructuras que lo forman así como también obtener la energía, esencial para desarrollar todas las actividades de la vida cotidiana [26]. Es sumamente importante comprender que nuestro cuerpo necesita de energía para su funcionamiento, la cual debe llegar a cada una de las células del organismo, lo que incluye el cerebro, los músculos, el aparato digestivo, la piel y el hígado [27].

La fuente principal de energía es obtenida a partir de la ingesta de hidratos de carbono (como el azúcar, pan, frutas, cereales y legumbres, entre otros), los cuales pueden ser de dos tipos: simples o complejos (almidones). Mientras que los azúcares simples se absorben inmediatamente en el torrente sanguíneo, los almidones deben ser transformados previamente, mediante el proceso de digestión, en unidades más pequeñas denominadas glucosa [8]. Cuando el organismo posee más glucosa de la necesaria, esta se absorbe y deposita en el hígado y músculos como glucógeno[1] (de forma limitada) [29]. El azúcar que no se almacena en el hígado será liberado en la corriente sanguínea y desde ahí se distribuirá a todas las células del cuerpo [4].

La obtención de energía por parte de nuestras células, actividad que desarrollan en todo momento del día, sería imposible sin la hormona que permite a estas captar las moléculas de glucosa. Esta hormona, denominada insulina, es secretada por el páncreas [4].

El páncreas es un órgano situado en la zona superior del abdomen detrás del estómago y junto al intestino delgado que, en su función endocrina, se encarga de producir

[1] Biomolécula que forma parte de los glúcidos, también llamados hidratos de carbono o carbohidratos, presente en el hígado y almacenado por el organismo a modo de reserva hasta que, llegado el momento de su utilización, lo convierte en glucosa [28].

hormonas sobre el torrente sanguíneo para influir sobre otra parte distante del organismo. Las porciones de páncreas que liberan estas hormonas se agrupan en islotes pancreáticos o de Langerhans. Dichos islotes contienen diferentes tipos de células con diferentes funciones; la encargada de la producción y liberación de insulina son las células β mientras que las células α se encargan de la liberación de otra hormona conocida como glucagón [30].

La Figura 2.1 resume el mecanismo de regulación de glucemia. Cuando ingerimos hidratos de carbono, el nivel de glucemia aumenta. Cuando la sangre pasa por el páncreas y las células β reconocen niveles de glucemia elevados, aumenta la liberación de insulina hacia el torrente sanguíneo. De esta manera, la insulina viaja por la sangre hasta encontrarse con los receptores insulínicos que se encuentran en la superficie de nuestras células. Una vez unida a ellos, la insulina induce a que los transportadores de glucosa de la célula que contenía en su interior se coloquen en la membrana, lo que permite el ingreso de glucosa hacia el interior de la célula. De esta manera, queda disponible para su uso posterior a fin de obtener energía [8].

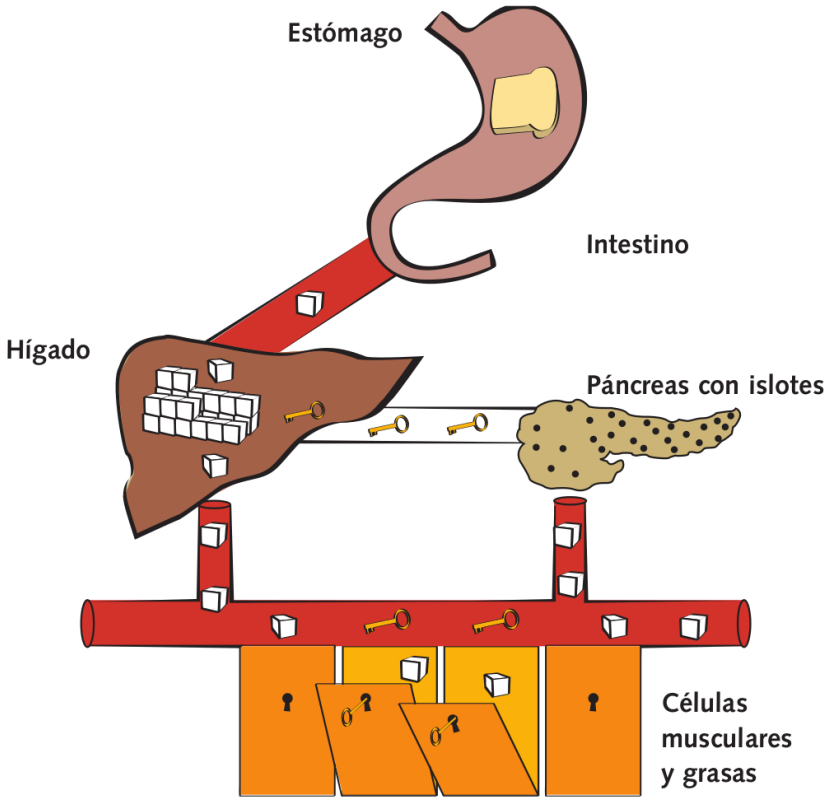


Figura 2.1: Proceso de metabolización de la glucemia. Extraído de [8].

Se podría pensar en que la insulina es la "llave" encargada de abrir las membranas o "cerraduras" de las células del organismo para el ingreso de glucosa circulante por la sangre.

Por otra parte, el glucagón actúa en el hígado, donde activa procesos metabólicos

que permiten degradar las reservas de glucógeno en glucosa para su liberación a la sangre. El glucagón y la insulina actúan de manera coordinada permitiendo mantener niveles normales de glucemia y asegurando a las células del organismo contar con suficiente fuente de energía para realizar sus funciones vitales. Esto representa la principal función del páncreas que es regular los niveles de glucemia, ya sea liberando insulina o glucagón cuando ésta se eleva o disminuye por demás, respectivamente. De esta forma, se logra mantener el nivel de glucemia dentro de los rangos normales en una persona no diabética [31]. En la Figura 2.2, se describe mediante un mapa conceptual la producción y acción de la insulina.

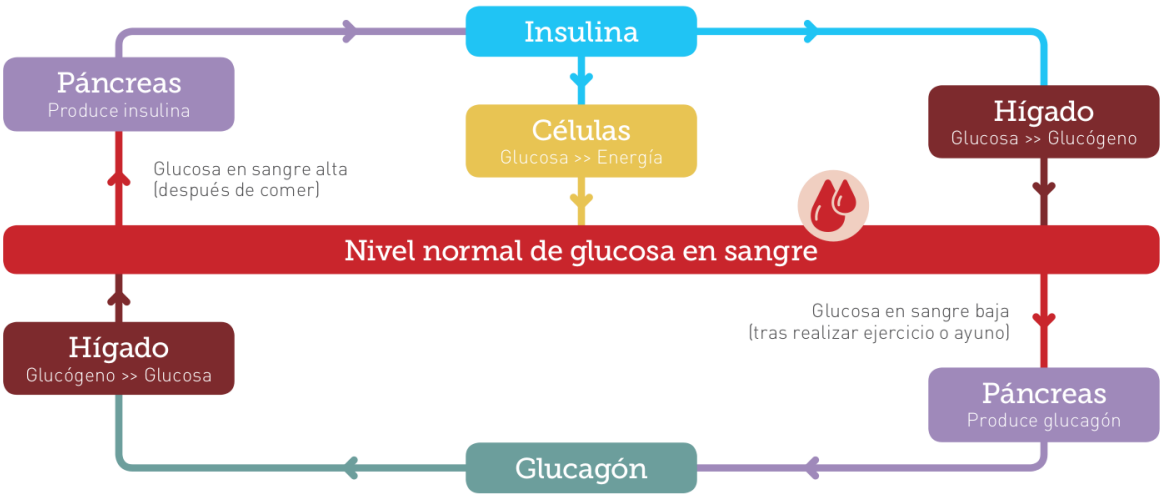


Figura 2.2: Producción y acción de la insulina. Extraído de [10].

En base a lo expuesto se puede comprender que la insulina es una hormona que forma parte de un sistema complejo con implicaciones a muchos niveles del cuerpo. En tanto, un fallo en el funcionamiento como en la producción de esta hormona puede tener consecuencias muy graves. Una de ellas es la DM, en la cual las células pierden la capacidad de captar y utilizar la glucosa como fuente de energía. En consecuencia, se produce un aumento sostenido del nivel de glucemia por encima del valor normal, condición conocida como hiperglucemia. Este fallo es causado en el páncreas, ya sea por producir insulina defectuosa, o bien, por perder directamente la capacidad de producir la cantidad suficiente de esta hormona. Los dos tipos principales de DM se diferencian justamente por lo anterior.

2.1.2 Tipos

La complejidad de esta afección hace a la existencia de múltiples tipos, cada uno con sus particularidades. Las principales formas clínicas de la DM son:

- La DT1 es la menos común de las dos, representando alrededor del 10% de las personas que padecen esta enfermedad [2]. También se la conoce como diabetes

juvenil ya que afecta principalmente a niños y adolescentes, aunque puede afectar a personas de cualquier edad. En este tipo de DM, las células β del páncreas están totalmente destruidas, por lo que no se produce insulina. La destrucción de estas células suele ser debido a una reacción autoinmune, en la cual el sistema de defensa del cuerpo las ataca por error. Como resultado, el cuerpo no produce insulina o produce una cantidad insuficiente de esta. Al no haber dicha hormona, las células no pueden absorber la glucosa (no existe la "llave"), y, por lo tanto, no son capaces de generar energía eficientemente. Por consiguiente, las células se debilitan y la concentración de glucosa en sangre se eleva llevando a la hiperglucemia. Debido a esto, las personas que padecen este tipo se las llama insulinodependientes ya que necesitan administrarse diariamente inyecciones de insulina para controlar el nivel de glucemia y así sobrevivir.

- La forma más frecuente es la DT2, la cual se encuentra representada por casi el 90% de los casos [3]. En gran medida, se debe al exceso de peso y a la falta de actividad física. En esta variante, el cuerpo es capaz de producir insulina pero se vuelve resistente a ella de a poco. A diferencia de la DT1, además del porcentaje de casos, este tipo afecta principalmente a personas adultas mayores aunque se ve cada vez más en niños y adolescentes por los niveles crecientes de obesidad, inactividad física y dieta inapropiada [4]. En la DT2, puede pasar que el páncreas no produzca suficiente insulina. También puede pasar que el páncreas produzca insulina pero las células del organismo resisten los efectos de la misma debido a que los receptores insulínicos son incapaces de reconocerla. Asimismo, tanto la resistencia como la deficiencia de insulina pueden ocurrir al mismo tiempo. En todos los casos, la insulina es ineficaz, es decir, no es capaz de actuar como llave para dejar ingresar la glucosa a las células, lo que se conoce como resistencia a la insulina. Consecuentemente, los niveles de glucemia terminan aumentando, lo que desencadena una hiperglucemia. Aunque pueda parecer contradictorio, esta condición estimula mucho a las células β del páncreas, quienes producen cada vez más insulina, llegando a un estado conocido como hiperinsulinemia. En algunas personas, con el tiempo y luego de tanto esfuerzo, las células β del páncreas se desgastan y pierden lentamente la capacidad de producir insulina adecuadamente, pasando a un estado de DM.

El factor común se encuentra representado por una hiperglucemia como consecuencia de una insuficiencia de insulina; sin embargo, el mecanismo de este aumento no es exactamente igual en los dos tipos de DM. En los pacientes con DT1, o insulinodependientes, el páncreas no fabrica, o al menos en cantidad suficiente, insulina, mientras que las personas con DT2, generalmente obesas, el páncreas secreta una cantidad mayor de insulina pero existe una resistencia o deficiencia de dicha hormona.

A veces las personas cuando tienen DM dicen que tienen "un poquito alto el azúcar" o que tienen PDM. Estos términos hacen pensar que la persona realmente no tiene DM o que su caso es menos grave. Sin embargo, todos los casos son graves [32]. El término de PDM, puede verse englobado dentro de la TGA, la GAA o una combinación de ambas

(TGA+GAA), donde los niveles altos de glucosa en sangre excede los límites normales pero se encuentran por debajo del umbral de diagnóstico de DT2 [4].

La relevancia de estos estados es cada vez mayor debido a que son estadios de transición entre la normalidad y la DM. Esto significa que pueden evolucionar hacia DT2 dependiendo de la gravedad (basada en los niveles de hiperglucemia) y otros factores de riesgo como la edad y el peso [33]. La importancia de la TGA y la GAA se debe a tres motivos: en primer lugar, la implicancia de riesgo para el desarrollo de DT2; en segundo lugar, denotan un alto riesgo de padecer enfermedades cardiovasculares; y, por último, su detección abre el camino a la adopción de intervenciones para prevenir la DT2 [4].

Si bien la presente tesina se encuentra enfocada en la identificación de DT2, por su carácter "silencioso" y predominancia, resulta importante mencionar que existen otros tipos. En [34] la Organización Mundial de la Salud (OMS) enumera "otros tipos específicos", aunque representan un porcentaje inferior al 2%.

2.1.3 Causas

Como otros fenómenos de la vida, el desarrollo de DM es el resultado de una compleja interacción entre una predisposición genética y factores de riesgo ambientales y de comportamiento. Lo más alarmante es el número de personas en rápido aumento que padecen esta enfermedad a nivel mundial [4]. Además de la importancia de la predisposición genética de la persona, se han identificado algunos factores de riesgo ambientales y/o infecciones virales que pueden contribuir al desarrollo de DT1 si se combinan con dicha sensibilidad genética. También se ha implicado a algunas toxinas o factores alimenticios [4]. Sin embargo, los factores de riesgo para la DT1 aún están siendo investigados desconociéndose con exactitud [2].

Asimismo, en la DT2 también existe una predisposición genética; de hecho, si una persona tiene un familiar con DM es probable que esa persona también termine desarrollando la afección. Aunque las causas exactas del desarrollo de la DT2 no se conocen en su totalidad aún, en este tipo, la influencia del ambiente y del estilo de vida es una cuestión comprobada.

La obesidad es la principal razón de la resistencia a la insulina y ésta, a su vez, es la principal responsable del desarrollo de DT2. Esto se debe a que en una persona obesa la insulina no puede actuar correctamente debido a una insulinorresistencia provocada por una alteración en los receptores insulínicos de las células. Si el páncreas aumenta la liberación de insulina, la glucemia permanecerá normal (obesidad sin DM). Sin embargo, en personas con predisposición genética a desarrollar DM, después de un tiempo el páncreas no liberará toda la insulina necesaria conduciendo a una DT2 [8]. La prevalencia de obesidad en la población, como producto de un ritmo de vida sedentario y dietas menos saludables, hace que dicha enfermedad se haya disparado exponencialmente en los últimos años.

Otros factores de riesgo que juegan un papel importante son el sexo, sobrepeso, la inactividad física, nutrición pobre, historial familiar de DM, historial pasado de diabetes gestacional, edad avanzada [10], consumo de alcohol [35], [36], tabaquismo [37], la duración del sueño [38], [39], enfermedades cardiovasculares y sus factores de riesgo y el

origen étnico (sur de Asia, afrocaribeño e hispano) [40]. Por su parte, la PDM también es considerada dentro de los factores de riesgo de DT2, representando un estadio previo a la misma. Por último, algunas mutaciones genéticas, otras enfermedades hormonales, la lesión del páncreas y ciertas medicinas también pueden causar DM [41].

2.1.4 Síntomas y diagnóstico

En un estado de DM, las personas poseen niveles de glucemia elevados. Como consecuencia, cuando el aumento del nivel de glucemia supera la capacidad del riñón, el cuerpo comienza a eliminar glucosa junto con agua por la orina. Este es el desencadenante de los dos síntomas clásicos de la DM como son la poliuria (orinar con frecuencia[2]) y la polidipsia (necesidad exagerada de beber agua). Por más de una ingesta de alimentos, las células son incapaces de captar los nutrientes y por tanto siguen necesitando energía, motivo por el que aparece otro síntoma llamado polifagia (aumento de ganas de comer).

Asimismo, la ausencia de insulina lleva a que las células no absorban los aminoácidos, lo que provoca la falta de proteínas. Como consecuencia, la persona se siente débil, pierde peso y sufre alteraciones en muchos tejidos del cuerpo.

Con esto, se identifican las cuatro "P" sintomáticas de una persona diabética: poliuria, polidipsia, polifagia y pérdida de peso. La presencia de estos síntomas deberían confirmarse mediante análisis bioquímicos que confirmen el diagnóstico de la enfermedad o la descarten.

Conocer las cuatro "P" de la DM permitirá prestar atención sobre la presencia en personas cercanas, de manera de tratar precozmente la enfermedad y así evitar las consecuencias que la DM no controlada puede provocar.

Aunque los síntomas principales están dados por los anteriormente mencionados, como consecuencia de una hiperglucemia también pueden aparecer las siguientes alteraciones:

- Dificultad para la cicatrización de las heridas.
- Infecciones frecuentes de la piel y vías urinarias.
- Picazón (prurito).
- Visión borrosa.
- Cansancio fácil, aún sin realizar trabajo físico intenso (astenia).

Los síntomas entre los diferentes tipos de DM suelen ser similares y se engloban dentro de los mencionados anteriormente. El comienzo de la DT1, en general, se da de forma brusca y con grandes manifestaciones clínicas por lo que se la diagnostica con facilidad [8]. Sin embargo, en ocasiones, algunos síntomas clínicos clásicos, como la

[2]Asociada con la etimología de la enfermedad, diabetes hace alusión al paso del líquido desde su ingestión hasta la micción, y, mellitus proveniente del latin mel "miel" referida al carácter de esa orina similar a la miel con su dulce matiz.

polidipsia, poliuria y pérdida de peso no se presentan y, por lo tanto, el diagnóstico se puede retrasar o incluso pasar por alto [4], [42]. Esto implica que en algunas ocasiones, el diagnóstico de la DT1 puede ser difícil, por lo que es posible que se requieran pruebas adicionales para confirmar el mismo [2].

Los síntomas de la DT2 pueden ser similares a los que ocasiona la DT1 aunque con frecuencia pueden ser leves o estar ausentes. El aumento de la glucemia en la DT2 se produce en forma lenta y progresiva. En consecuencia, las personas pueden vivir varios años con esta afección antes de ser diagnosticada [3], [42]. Cuando no se identifica la enfermedad por un tiempo prolongado, en el momento del diagnóstico pueden estar ya presentes ciertas complicaciones. Por este motivo, es difícil determinar con precisión el momento exacto de aparición de la enfermedad [8]. Más aún, se estima que entre un tercio y la mitad de las personas con DT2 no reciben el diagnóstico correspondiente a tiempo [4]. Los pacientes pueden encontrarse en un estado de PDM por 20 años hasta llegar a la DT2.

El diagnóstico de la DM puede realizarse por medio de análisis de sangre. Cualquier persona con presencia de síntomas o factores de riesgo asociados a DM debe ser examinada. Existen distintas pruebas que permiten detectar tanto DM como PDM tempranamente y así trabajar sobre ellas para prevenir sus complicaciones. Así, los cambios del estilo de vida encaminados a perder una cantidad moderada de peso, si se tiene sobrepeso, pueden ayudar a retardar o prevenir la DT2 [43].

Dentro de las pruebas utilizadas para el diagnóstico de DM y PDM se encuentran:

- *Glucosa plasmática venosa o capilar en ayunas*. Mide el nivel de glucosa en la sangre en un momento concreto. Para conseguir los resultados más fiables, habitualmente se realiza el examen por la mañana en ayuno[3]. Si el laboratorio no puede realizar una extracción en forma venosa, los dispositivos que reportan los valores de glucosa en el plasma capilar[4] pueden ser utilizados. Esta prueba es menos costosa pero tiene la dificultad de garantizar el estado de ayuno.

- *Prueba A1C*[5]. Permite conocer el grado de compensación de la DM durante los últimos 2 a 3 meses. Otros nombres para la prueba A1C son prueba de la hemoglobina A1c, HbA1C y hemoglobina glucosilada. La determinación HbA1c mide el porcentaje de hemoglobina (pigmento rojo de la sangre) que tiene glucosa adherida a su estructura. Los resultados de la prueba A1C se muestran en forma de porcentaje. Cuanto más alto sea el porcentaje, más alto será el promedio de los niveles de glucosa en la sangre en los últimos tres meses [8]. Esta prueba no requiere ayuno, aunque es más costosa que las pruebas de glucosa. También, en algunas afecciones como la anemia o insuficiencia renal, puede ser inexacta [40].

[3]Ausencia de ingesta calórica entre 8 y 14 horas.

[4]La glucemia capilar se mide mediante un pequeño pinchazo en un dedo para extraer una gota de sangre que luego se coloca en una tira reactiva y se analiza mediante un glucómetro [44].

[5]Según la OMS, esta prueba se debe realizar en un laboratorio que aplique el método certificado por el NGSP y estandarizado para el Ensayo sobre el control y las complicaciones de la diabetes.

- *Prueba de glucosa plasmática aleatoria (o glucemia aleatoria)*. Solo debe ser utilizada en presencia de síntomas y no se desea esperar a que la persona tenga ayuno, es decir, puede ser utilizada en cualquier momento.

- *Prueba de tolerancia oral a la glucosa (PTOG)*. Mide la capacidad del cuerpo de utilizar la glucosa luego de beber una cantidad estándar de líquido que contiene la misma. Se requiere guardar ayuno antes de esta prueba. Primero, se toma una muestra de sangre en ayunas y luego se da a beber un líquido que contiene cierta cantidad de glucosa[6]. Para diagnosticar DM se deberá volver a medir el nivel de glucosa sanguínea periódicamente durante las siguientes dos o tres horas a fin de determinar la rapidez con la que la glucosa se elimina de la sangre. Luego de cierto tiempo, los niveles de insulina y glucosa en la sangre deberían volver a la normalidad; de no ser así, se podría estar en presencia de DM o algún estadio previo. Esta prueba ayuda a detectar la DT2 y PDM de forma más efectiva que la prueba de glucosa plasmática en ayunas, aunque, es una prueba más costosa y no es sencilla de realizar.

Los umbrales de corte para el diagnóstico de DM y PDM según la OMS se encuentran definidos en la Figura 2.3. La mayor parte de las guías aplican estos criterios estándar de diagnóstico propuestos por la OMS.

[6]Según la OMS, esta prueba se debe realizar con una solución glucosada que contenga el equivalente a 75g de glucosa anhidra disuelta en agua.

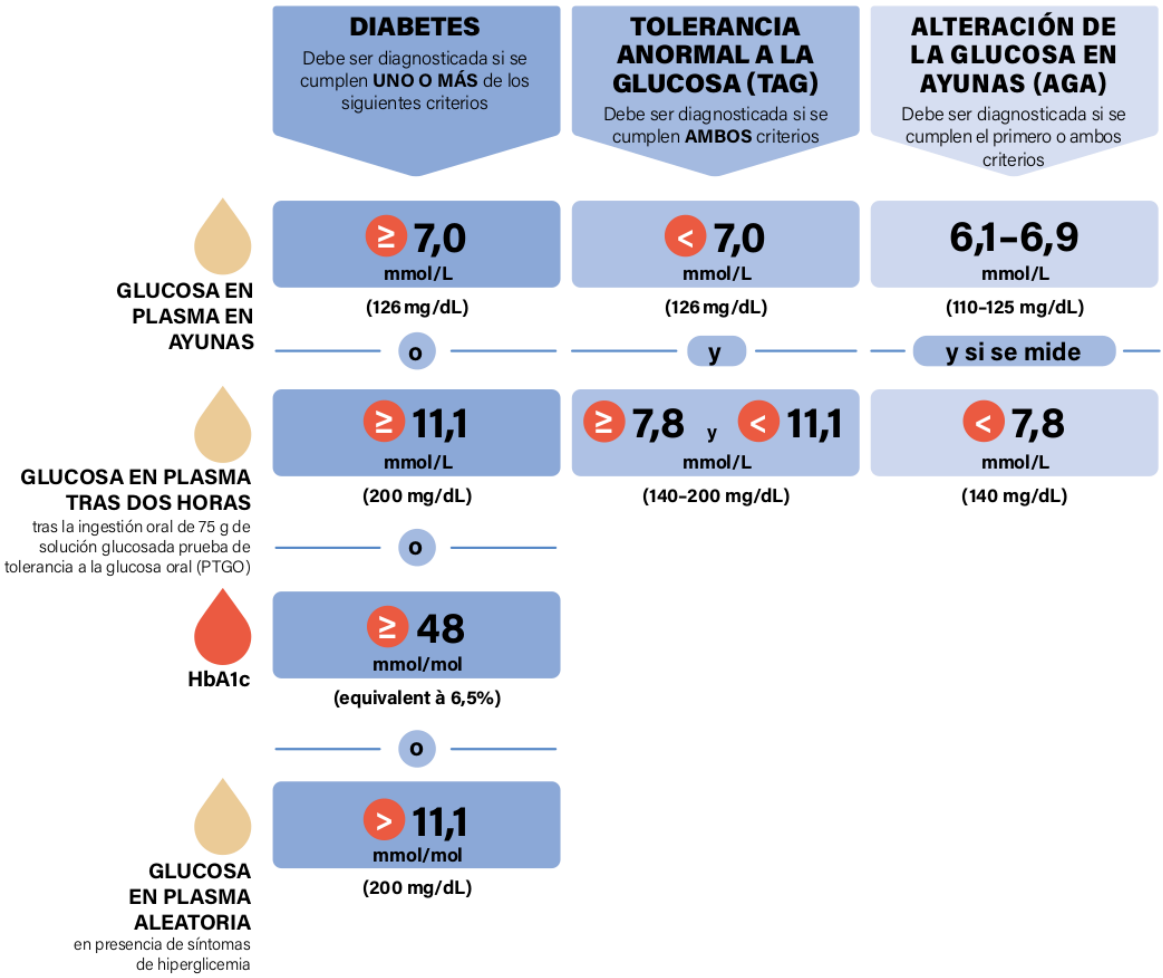


Figura 2.3: Criterios de diagnóstico para la diabetes. Extraído de [4].

Aunque estas pruebas y umbrales pueden diagnosticar DM, no permiten identificar su tipo. Algunas veces la identificación del tipo de DM es difícil y se requieren pruebas adicionales para distinguir entre la DT1 y la DT2, u otros tipos de DM. Como veremos más adelante, en todos los casos, el tratamiento de la enfermedad es dependiente de su tipo y, por lo tanto, es importante saberlo.

2.1.5 Complicaciones

En las últimas décadas, la DM se ha consolidado como uno de los problemas sociosanitarios de mayor importancia a nivel mundial en función de su elevada frecuencia (incidencia y prevalencia), las frecuentes y graves complicaciones tardías de la enfermedad, el impacto que éstas provocan sobre la calidad de vida del paciente, los costos económicos y sociales y, por último, su elevada tasa de mortalidad. Asimismo el impacto de las complicaciones derivadas de la DM sobre el entorno social, familiar y laboral de los enfermos es cada vez más importante.

A nivel humano, la DM tiene un costo elevado. Dicha afección y sus complicaciones son las principales causas de muerte en la mayoría de los países a nivel mundial [4], [10]. Además de suponer una gran carga económica, no solo para los individuos sino también para sus familias, debido al costo de la insulina y otros medicamentos esenciales para su tratamiento y control, la DM también tiene un impacto económico sustancial para los países y los sistemas sanitarios nacionales. Esto es debido a un mayor uso de los servicios de salud, pérdida de productividad y el apoyo a largo plazo necesario para superar las complicaciones relacionadas con la enfermedad. La mayoría de los países gastan entre un 5% y un 20% del total del gasto sanitario en DM [10]. Más aun, las inquietudes personales sobre la aparición de complicaciones futuras y su posible impacto en la calidad de vida también contribuyen de manera significativa a los costos intangibles de la afección: los que surgen de la preocupación, la ansiedad, la incomodidad, el dolor, la pérdida de la independencia y una infinidad de factores no económicos pero con una importancia fundamental para la vida con DM. Con costos tan elevados, la enfermedad supone un desafío significativo para los sistemas sanitarios y un obstáculo para el desarrollo económico sostenible.

Potencialmente esta afección puede conducir a numerosas complicaciones que debilitan la salud, disminuyen la calidad de vida e incluso causan una muerte temprana. A menudo, las complicaciones son producto de una DM sin tratar o de un tratamiento inapropiado.

De acuerdo a estimaciones realizadas en muchos países por la FID, aproximadamente el 50% de las personas con DM no son conscientes de que viven con ella y, por lo tanto, poseen una alta probabilidad de mantenerse sin diagnosticar hasta la aparición y/o progresión de las complicaciones asociadas.

Las complicaciones agudas de la DM, o a corto plazo, originadas a raíz de niveles extremos de glucosa en sangre son frecuentes en la DT1 aunque, con determinados medicamentos, pueden ocurrir en la DT2 y otros tipos de DM. Estas complicaciones pueden derivar en consecuencias neurológicas permanentes o incluso la muerte. Por esta razón, estas condiciones pueden y deben evitarse.

Como la elevación de la glucemia no produce dolor, suele ser imposible determinar el momento exacto del origen de la DT2 [8]. En consecuencia, cuando no se identifica la enfermedad por un tiempo prolongado, en el momento del diagnóstico es posible que ya presente complicaciones crónicas o tardías. Estas complicaciones también pueden aparecer poco después del inicio de una DT1 o bien por un control deficiente.

Cuanto más tiempo se mantiene el estado de hiperglucemia, mayor es el número de complicaciones tardías que pueden aparecer, dañando de manera lenta y progresiva distintos tejidos y órganos del cuerpo. Esto da lugar al desarrollo de diversas alteraciones que conllevan a las siguientes complicaciones para la salud:

- *Enfermedades de los ojos*. Los persistentes niveles altos de glucosa en sangre son la principal causa de retinopatía. La red de vasos sanguíneos que irrigan la retina puede dañarse a causa de la retinopatía, dando lugar a la pérdida permanente de visión. La prevalencia de la retinopatía diabética aumenta con la duración de la

DM, tanto de DT1 como de DT2, y se asocia con un control glucémico deteriorado y la presencia de hipertensión.

- *Enfermedad cardiovascular*. Las Enfermedades Cardiovasculares (ECV) son la causa más común de muerte e incapacidad entre las personas con DM. Las ECV que acompañan a la DM incluyen la angina de pecho, infarto de miocardio (ataque al corazón), accidente cerebrovascular, enfermedad arterial periférica y la insuficiencia cardíaca congestiva. Altos niveles de presión arterial, colesterol, nivel de azúcar en sangre, así como otros factores de riesgo contribuyen al aumento del riesgo de complicaciones cardiovasculares. Las ECV representan entre un tercio y la mitad del total de muertes relacionadas con la DM [45]. En personas con DT2, el riesgo de tener un infarto de miocardio es mayor que en personas sin DM.

- *Complicaciones en el embarazo*. Las mujeres con cualquier tipo de DM corren riesgo de complicaciones durante el embarazo, ya que los altos niveles de glucosa pueden afectar el desarrollo del feto y conducir a problemas durante el parto, lesiones en el niño y la madre, e hipoglucemia en el niño tras el nacimiento. Los niños que están expuestos a niveles altos de glucosa en sangre en el útero tienen un mayor riesgo de desarrollar DT2 en el futuro.

- *Pie diabético*. Además del daño a los nervios, las personas con DM pueden experimentar problemas con la mala circulación en los pies como resultado del daño en los vasos sanguíneos. Estos problemas aumentan el riesgo de ulceración, infección y amputación. Las personas con DM se enfrentan a un riesgo de amputación que puede ser 25 veces mayor que las personas sin DM [10]. El pie diabético y las complicaciones en los miembros inferiores afectan a entre 40 y 60 millones de personas con DM en todo el mundo [4].

- *Enfermedad renal*. La DM es una de las causas principales de la enfermedad renal crónica (nefropatía), siendo mucho más común que en personas sin DM. La enfermedad es originada por el daño a los vasos sanguíneos pequeños, que puede causar que los riñones sean menos eficientes, o fallen por completo.

- *Daño en el sistema nervioso*. El daño en los nervios (neuropatía) también es el resultado de niveles de glucosa altos prolongados, pudiendo afectar a cualquier nervio en el cuerpo. El tipo más común es la neuropatía periférica, que principalmente afecta a los nervios sensoriales en los pies y puede producir dolor, hormigueo y pérdida de la sensibilidad. Esto es particularmente importante porque puede permitir que las lesiones pasen desapercibidas, lo que lleva a la ulceración, infecciones graves y en algunos casos amputaciones. En la actualidad, menos de un tercio de los médicos reconocen los síntomas de la neuropatía periférica relacionada con la DM, lo que contribuye enormemente a los altos índices de morbilidad y mortalidad de esta complicación [45].

La frecuencia y precocidad de estas complicaciones se asocia a la edad de inicio de la enfermedad y al control inadecuado de ésta. Asimismo, las consecuencias y evolución de

la DM no son iguales en todos los casos y dependen, entre otros factores, de la edad del paciente.

2.1.6 Prevalencia e impacto

A pesar de los múltiples esfuerzos por conocer cada vez más acerca de esta enfermedad y promover tratamientos para su confrontación, la prevalencia mundial de DM se ha disparado en los últimos años a punto tal que algunos la han llegado a nombrar la "epidemia del siglo XXI" [46].

En 2019 la FID confirmó que la DM es una de las emergencias de salud que crece de manera vertiginosa alcanzando niveles alarmantes: casi 500 millones de personas a nivel mundial vivían con esta enfermedad. Más aún, se estimó que esta cifra seguiría en aumento alcanzando los 700 millones para 2045 [4]. Otro factor alarmante, a nivel de Latinoamérica, es el creciente porcentaje de personas con DM sin diagnosticar, que ya en 2019 alcanzaba el 42% (13 millones) de adultos entre 20 y 79 años [4]. En ese entonces, 1 de cada 3 personas con DM no se encontraba diagnosticada y 1 de cada 11 adultos padecían DM [47]. Debido al crecimiento de tipo epidémico de esta afección, en el año 2017, la Organización Panamericana de la Salud, presentó en [48] la inclusión de la DM entre sus prioridades dentro de las enfermedades no transmisibles para una vida sana y productiva.

Según la FID, tres de cada cuatro personas que viven con DM están en edad activa (es decir, entre 20 y 64 años) previendo un aumento marcado que llegaría a los 486 millones para 2045. Por otra parte, China, India y Estados Unidos son los países con el mayor número de adultos con DM, y se pronostica que esta situación se mantendrá hasta el año 2030. Se prevé que para 2045 el número de personas con DM en Pakistán superará al de Estados Unidos, lo que situará al país en el tercer lugar [45].

Está claro que la DM tiene una importante prevalencia mundial ya que representa una gran amenaza para la salud y, en ese sentido, Argentina no es la excepción. En 2018, la DM fue responsable de 9.086 muertes con un 90% producidas en edades superiores a los 55 años [49]. Según la 4ta Encuesta Nacional de Factores de Riesgo, realizada en 2018, la prevalencia de glucemia elevada o DM por autoreporte en la población adulta aumentó de 9,8% en 2013 a un 12,7%, en concordancia con el exceso de peso, uno de los principales factores de riesgo de la DM. En tanto, la prevalencia combinada de glucemia elevada o DM en mayores de 18 años es de 10,9%. Entre quienes no se autoreportaron como diabéticos ni refirieron jamás haber tenido una glucemia elevada, el 5% tuvo la glucemia elevada y 20% de la población presenta alto o muy alto riesgo de desarrollar DT2 a 10 años [5].

Estos datos ayudan a comprender y concientizar acerca del impacto de esta enfermedad a nivel mundial y, por lo tanto, la necesidad imperiosa de utilizar herramientas que faciliten su detección temprana a fin de lograr un tratamiento apropiado.

2.1.7 Tratamiento

Notablemente, gran parte de la carga socioeconómica y de enfermedad mencionada anteriormente puede prevenirse en diferentes niveles mediante distintos tratamientos y programas ampliamente estudiados y discutidos en la literatura médica especializada. Si bien, la DM por ahora no tiene cura, la enfermedad puede compensarse perfectamente, permitiéndole al paciente llevar una vida prácticamente normal y sana [32].

A fin de lograr un control adecuado de los niveles de glucemia, las personas con DM necesitan tratamiento permanente y a largo plazo [8]. El paciente debe participar activamente en el control y tratamiento de su enfermedad.

Afortunadamente, existe consenso a nivel nacional y global [4], [8], [50], [51] sobre cómo lograr un tratamiento exitoso de esta enfermedad, el cual se apoya en cuatro pilares: educación, plan de alimentación saludable, práctica regular de actividad física y diversos medicamentos (comprimidos hipoglucemiantes orales e insulina).

La educación diabetológica es el pilar fundamental que permite la participación eficaz del paciente durante el control y tratamiento de su enfermedad. Se inicia en el consultorio junto al médico diabetólogo para luego complementarse con lecturas de información variada y cursos de formación específicos [8].

En el tratamiento, la alimentación saludable, saber qué y cuándo comer, juega un papel muy importante. Realizar un plan de alimentación saludable ayudará a controlar el nivel de glucemia, la presión arterial y el colesterol.

La actividad física regular es un hábito muy saludable para todas las personas, y, aún más, en aquellas con DM por favorecer al buen control metabólico de su enfermedad: disminuye la glucemia, ayuda a bajar de peso, y potencia y mejora la acción de la insulina [8]. Sin embargo, el aumento de la actividad física debe estar controlado y adecuado a las características de cada persona, para que no resulte contraproducente.

Las personas con DT2 inicialmente son capaces de producir insulina, por lo que en principio son tratadas con un plan de alimentación y actividad física regular y adecuada al paciente. Si este plan inicial no consigue regular la glucemia adecuadamente, entonces deberá iniciarse el tratamiento complementario con medicación o insulina a fin de lograr un control adecuado de la enfermedad [8]. Esta situación no escapa a pacientes con DT1.

Existen varios medicamentos, como las sulfonilureas, la metformina y las tiazolidinedionas, que permiten controlar los niveles de glucemia. Cada uno de los distintos tipos de comprimidos hipoglucemiantes actúan de distintas formas y son útiles en función del tipo de DM y su estado de evolución [10].

Las metas para el tratamiento se establecen selectivamente para cada paciente, tipo de DM y gravedad de la enfermedad. Cada caso en particular deberá recibir un plan de alimentación, actividad física y medicación individualizada acorde a las metas establecidas.

A falta de producción propia de insulina, el paciente con DT1 debe administrarse la hormona a diario normalmente a través de inyecciones. Es esencial que todas las personas con DT1 tengan un suministro ininterrumpido de dicha hormona, realicen un control regular de la glucemia y adopten un estilo de vida saludable para controlar su afección de manera eficaz. A pesar de cumplir con las metas del tratamiento, en

algunos pacientes con DT2 la glucemia puede seguir manteniéndose elevada. Esto suele ocurrir con un grado mayor de evolución de la enfermedad donde el páncreas ya no produce insulina o al menos en cantidad suficiente. En estos casos también es necesario administrar insulina a fin de controlar los niveles de glucemia.

En general, los pacientes con DM, deberán realizar un chequeo diario del nivel de glucemia para controlar su enfermedad, en aquellos con DT1 (o quienes se administren insulina) este control debe ser más estricto y frecuente. La Automonitorización de la Glucosa en Sangre (AMGS) es el nombre que recibe el proceso de medición de glucemia realizado por personas con DM en forma diaria. La AMGS ayuda a las personas con DM y médicos a comprender cómo varían sus niveles de glucemia durante el día para que su tratamiento se pueda ajustar en consecuencia a fin de obtener los mejores resultados. Los resultados de cada monitoreo ayudarán a tomar decisiones sobre la alimentación, actividad física y los medicamentos.

Actualmente, no existe una intervención eficaz y segura para prevenir la DT1 a pesar de una gran cantidad de ensayos clínicos destinados a detener y proteger la destrucción autoinmune en curso de las células β pancreáticas. Aunque no se han logrado resultados convincentes hasta el momento, es probable que sean objetivos más alcanzables en el futuro previsible. En contraposición, las investigaciones indican que la mayoría de los casos de DT2 y PDM podrían prevenirse mediante una dieta saludable y actividad física regular [4], [52], [53].

2.2 Aprendizaje Automático

En los últimos años, el ML dominó el campo de la IA con un marcado crecimiento exponencial, convirtiéndose en la rama más productiva e importante. Su extensión parecería no tener límites, apareciendo en distintos sectores empresariales, disciplinas de la ciencia, innumerables artículos académicos [54], e incluso, siendo parte de la vida cotidiana de muchas personas (aún cuando no lo sepan). A continuación, se abordará esta sección (2.2.1) realizando un repaso de las definiciones de IA, sus orígenes y surgimiento de ML. Luego, se detallarán los distintos tipos de aprendizaje (2.2.2) y el proceso de ML (2.2.3). Posteriormente, se describirán los modelos de ML utilizados en la presente investigación (2.2.4). Finalmente, se abordarán las métricas utilizadas con fines evaluativos y comparativos de los distintos modelos de ML (2.2.5).

2.2.1 Historia

Intentar dar una definición exacta de IA es una tarea compleja, sobre todo porque es un concepto que depende de la propia definición de inteligencia, que al día de hoy sigue teniendo múltiples interpretaciones. Debido a esto, existen muchos autores que la definen a su manera. Una definición concisa sería: *"Disciplina dedicada al estudio y diseño de sistemas que puedan imitar comportamientos inteligentes"*. Estos comportamientos pueden ser muy diversos: Conducir, analizar patrones, reconocer voces o ganar juegos. Son numerosas las formas en las que un sistema puede simular un comportamiento inteligente, habiendo cada vez más ejemplos de cómo, en ciertas áreas, logran alcanzar un rendimiento mayor al humano [55]–[57]. Un sistema "inteligente" es aquel que interpreta correctamente datos, aprende de ellos y emplea los conocimientos obtenidos realizando acciones que maximicen sus posibilidades de éxito en tareas concretas de forma adaptativa [58], [59].

Sin embargo, el nacimiento de la IA se produjo décadas antes de esta definición, cuando Alan Turing, en su artículo [60], comenzó a preguntarse si las computadoras tenían la capacidad de "pensar", pregunta que en la actualidad sigue abriendo nuevas investigaciones. En esos tiempos, el campo generó un gran optimismo por haber abordado y resuelto rápidamente problemas intelectualmente difíciles para los humanos pero simples para las computadoras. Cualquier problema que pudiera ser expresado a través de una lista de reglas explícitas formales, podía ser resuelto [61]. Este enfoque se conoce como IA simbólica y fue el paradigma dominante en la IA desde la década de 1950 hasta finales de la década de 1980 [61]. Bajo este enfoque, los programadores ingresan reglas (programa) y datos para ser procesados de acuerdo a las mismas a fin de obtener respuestas. Sin embargo, el verdadero desafío para la IA, consistía en resolver aquellos problemas complejos y confusos que son fáciles para los humanos, por su resolución intuitiva, pero difíciles de describir formalmente, como el reconocimiento de voz o identificación de rostros en imágenes [62].

Por el contrario, en lugar de que los programadores elaboraran reglas, ¿Podría una computadora aprenderlas automáticamente al observar los datos? ¿Podría una computadora ir más allá de "lo que sabemos cómo ordenarle que realice" y aprender por sí

misma cómo realizar una tarea específica? Así fue como surgió una nueva rama de la IA que logró su dominio posteriormente: el Aprendizaje Automático (ML, por sus siglas en inglés). Como se muestra en la Figura 2.4, con ML, los programadores ingresan datos, así como las respuestas esperadas a fin de obtener reglas que luego pueden ser aplicadas a nuevos datos para obtener respuestas originales [61]. De esta forma, un sistema de ML es entrenado en lugar de programarse explícitamente. Durante el entrenamiento, se presentan muchos ejemplos o experiencias para una tarea dada con el objetivo de "aprender" un patrón estadístico que permita generar reglas para automatizar la misma.

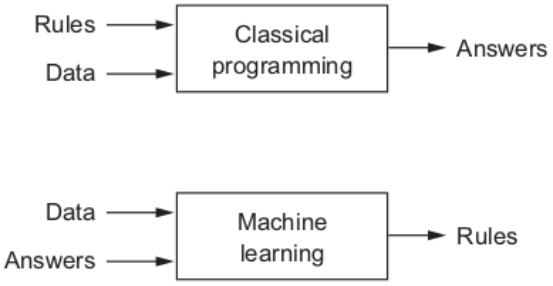


Figura 2.4: Aprendizaje Automático: un nuevo paradigma de programación. Extraído de [61].

El término de ML fue acuñado en 1959 por Arthur Samuel, el cual lo definió como: *"Campo de estudio que brinda a las computadoras la capacidad de aprender sin ser explícitamente programadas"* [12]. Por su parte, Tom Michael citó una definición más formal y abarcativa de ML en [63]: *"Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna clase de tareas T y una medida de performance P, si su desempeño en las tareas en T, medido por P, mejora con la experiencia E"*. Aquí es posible identificar un conjunto de objetos que permiten definir ML:

- Tarea (T), ya sea una o más.
- Experiencia (E).
- *Performance* (P).

De esta forma, con un sistema ejecutando un conjunto de tareas, la experiencia debería conducir a mejoras en el rendimiento [64]. Una tarea T consiste en aquellas dedicadas a resolver un problema del mundo real como la clasificación, la regresión, etc. La experiencia E se obtiene a partir del proceso de aprendizaje dado por el entrenamiento de los algoritmos de ML a través de un conjunto de datos o *dataset*. Esto conduce a la generación de un modelo de ML sobre el cual es posible determinar una métrica o medida cuantitativa (*performance* P) de la capacidad de generalización del conocimiento para realizar la tarea T.

Aunque el ML comenzó a generar interés en la década de 1990, se convirtió en los últimos años en el campo más popular y exitoso de la IA [65]. Esta tendencia se vio impulsada por el aumento significativo de la cantidad y calidad de información en conjunto con el desarrollo de *hardware* capaz de realizar un procesamiento eficiente de grandes volúmenes de datos a una escala sin igual [66]. A continuación, se presentan algunos ejemplos de aplicabilidad de ML que no hacen más que ratificar su popularidad:

- *Reconocimiento de voz y asistentes virtuales*. Capacidad que utiliza el Procesamiento de Lenguaje Natural (NLP, por sus siglas en inglés) para procesar el habla humana en un formato escrito. Muchos dispositivos móviles incorporan reconocimiento de voz en sus sistemas para realizar búsquedas por voz, o proporcionar más accesibilidad a los mensajes de texto [64]. Los asistentes virtuales, como *Siri* [67], *Alexa* [68] o *Google Now* [69], utilizan reconocimiento de voz para analizar la frase y orden de las palabras en ella a fin de generar una acción resultante como llamar a un contacto o enviar un mensaje de texto.

- *Visión por computadora*. Técnicas desarrolladas con el fin de interpretar y procesar imágenes digitales, videos y otras entradas visuales. Se intenta imitar (o mejorar) el sistema de visión humano. Entre otras áreas, la visión por computadora utiliza ML para generar modelos capaces de reconocer objetos en imágenes o videos. Sus aplicaciones son variadas, como el etiquetado de fotografías en redes sociales, identificación de automóviles [70], imágenes de radiología en el cuidado de la salud [71] y automóviles autónomos [72].

- *Sistemas de recomendación*. Los motores de recomendación detrás de las sugerencias de *Netflix* [73] y *Youtube* [74], la información del *feed* de *Facebook* [75] e *Instagram* [76], y la recomendación de productos se basan en ML [77]. Utilizando datos de comportamiento anteriores, los algoritmos de ML pueden ayudar a descubrir tendencias de datos que se pueden usar para llevar a cabo una acción, como desarrollar estrategias de venta cruzada más efectivas, recomendar un video o película o mostrar ciertas publicaciones o anuncios.

- *Medicina y salud*. La carrera está en marcha para que el ML se utilice en el análisis de la atención médica. Varias empresas están analizando las ventajas de usar el ML con *Big Data* para proporcionar a los profesionales de la salud datos mejor informados que les permitan tomar mejores decisiones [64]. *Watson for Oncology* [78] es una tecnología cognitiva que recomienda a los médicos opciones de tratamiento basadas en pruebas. Aunque es bastante fácil analizar datos de salud, el debate sobre la privacidad será el factor decisivo sobre cómo se utilizarán los algoritmos en última instancia [64].

2.2.2 Tipos

La premisa básica del aprendizaje a partir de datos es el uso de un conjunto de observaciones para descubrir un proceso subyacente. Sin embargo, esta es una visión muy

amplia y difícil de encajar en un marco único. Como resultado, han surgido diferentes paradigmas de aprendizaje para hacer frente a diferentes situaciones y supuestos.

- *Aprendizaje supervisado*. Cuando existe un conjunto de datos etiquetados de los cuales se conoce para cada ejemplo la clase a la que pertenece, entonces estamos dentro del entorno de aprendizaje supervisado [79]. La clase o *target* es el objetivo el cual se desea alcanzar. De esta manera, se proporciona un conjunto de datos de entrenamiento (variables de entrada) con las respuestas correctas (*targets*) y, en base a este, el modelo intenta aprender a generalizar el conocimiento buscando una relación existente (función) entre las entradas y salidas. Luego, el modelo puede ser capaz de realizar predicciones sobre futuras entradas desconocidas. Esto también se conoce como "aprender de los ejemplos" [80]. Los dos tipos de problemas principales que puede abarcar este tipo de aprendizaje son:

    * Problemas de clasificación, en donde la predicción del modelo refiere a una (o múltiples) etiquetas de clase para cada uno de los datos de entrada. El punto más importante sobre el problema de clasificación es que es discreto: cada ejemplo pertenece precisamente a una clase y el conjunto de clases cubre todo el espacio de salida posible. El objetivo será entonces encontrar la frontera o límite de decisión que puedan usarse para separar las diferentes clases [80]. Existe una amplia variedad de modelos pertenecientes a esta familia, los cuales abordaremos más adelante en la Sección 2.2.4, como Regresión Logística, K vecinos más cercanos, Árbol de Decisión, *Random Forest* y Redes Neuronales Artificiales. Principalmente, existen dos tipos de clasificaciones:
        - *Clasificación binaria*. Cuando sólo se tiene dos categorías o clases para los datos dados. Por ejemplo, el problema de clasificación de correo no deseado.
        - *Clasificación multiclase*. Cuando se tiene más de dos categorías o clases a las que el dato puede pertenecer. Por ejemplo, el problema reconocimiento óptico de caracteres.

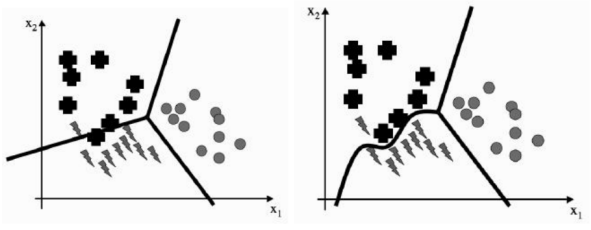


Figura 2.5: Problema de clasificación multiclase. Izquierda: Frontera de decisión en modelo lineal. Derecha: Frontera de decisión con una mejor discriminación aunque requiere un modelo no lineal. Extraído de [80].

    * Problemas de regresión, en donde el resultado de la predicción del modelo es un valor o cantidad continua. A nivel estadístico, un problema de regresión

consiste en ajustar una función que describa una curva, de modo que la misma pase lo más cerca posible de todos los puntos de datos [80]. Por ejemplo, el problema de predecir la edad de una persona dada su foto. Al igual que en clasificación, existe una amplia variedad de modelos de regresión, los cuales abordaremos más adelante en la Sección 2.2.4, como Regresión Lineal, K vecinos más cercanos, Árbol de Decisión, *Random Forest* y Redes Neuronales Artificiales.

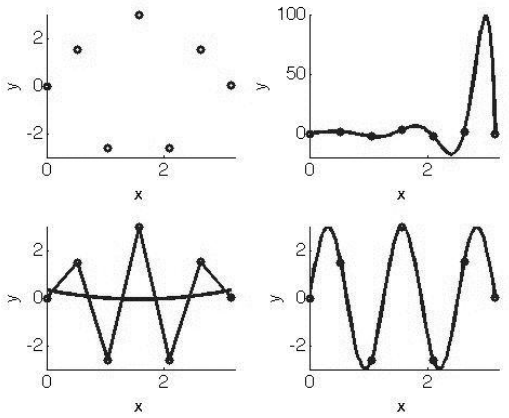


Figura 2.6: Problema de regresión. Arriba a la izquierda: Algunos puntos de datos de un problema. Abajo a la izquierda: Dos formas posibles de predecir los valores entre los puntos de datos conocidos: Conectando los puntos con líneas rectas o usando una aproximación cúbica (que en este caso pierde todos los puntos). Arriba y abajo a la derecha: Dos aproximadores más complejos que pasan por los puntos. Extraído de [80].

- *Aprendizaje no supervisado*. En este aprendizaje, el conjunto de datos de entrada no se encuentra etiquetado, es decir, solo se conocen los datos de entrada y no se proporciona ningún dato de salida. Este tipo de aprendizaje puede verse como la tarea de encontrar patrones y estructuras de forma espontánea en los datos de entrada [79]. Así, el modelo intentará aprender a partir del conjunto de entrada las relaciones, patrones, estructuras internas o características similares inherentes a los datos de forma tal que aquellos que tengan algo en común se categoricen juntos. De esta manera, en lugar de realizar predicciones sobre datos de entrada, se tiene un carácter exploratorio donde se intenta extraer conocimiento o información útil de los mismos. Asimismo, los algoritmos no supervisados son utilizados, entre otras cosas, para reducir la cantidad de características o *features* en un modelo a través del proceso de reducción de dimensionalidad; el Análisis de Componentes Principales (PCA, por sus siglas en inglés) y la Incrustación de Vecinos Estocásticos Distribuidos en t (t-SNE, por sus siglas en inglés) son enfoques comunes para esto.

- *Aprendizaje por refuerzo*. En algunas aplicaciones, la salida del sistema es una secuencia de acciones. En tal caso, una sola acción no es importante, más bien lo es la política, que es la secuencia de acciones correctas para llegar a la meta buscada.

Una acción es buena si es parte de una buena política. En tal caso, el modelo debería poder evaluar la bondad de las políticas y aprender de las secuencias de buenas acciones pasadas para poder alcanzar el desempeño deseado. Esto es el aprendizaje por refuerzo [66].

El aprendizaje automático supervisado es uno de los tipos más utilizados y exitosos, y será el enfoque empleado por los diferentes modelos de la presente investigación [66]. Por tal motivo, resulta preciso dar una definición formal del mismo obtenida de [81]: *"Dado un conjunto de entrenamiento de N pares de entrada-salida,* $\{(x_1, y_1), (x_2, y_2), ...(x_n, y_n)\}$*, donde cada* y_j *fue generada por una función* $y = f(x)$ *desconocida, el objetivo será descubrir una función h tal que se aproxime a la verdadera función f."*.

Tanto las x_i (datos de entrada) como las y_j (etiqueta o clase) pueden ser cualquier valor, no necesitan ser números. La función h también se llama hipótesis. El algoritmo de ML realizará una búsqueda sobre el espacio de posibles funciones válidas, llamado espacio de hipótesis, a fin de hallar una función que aproxime a la verdadera distribución de datos dada por f, es decir, que sus predicciones $\hat{y} = h(x)$ sobre los x_i del conjunto de entrenamiento se acerquen lo más preciso posible a las correctas dadas por $y = f(x)$.

No es de esperar que el aprendizaje reproduzca perfectamente la verdadera función. A fin de cuantificar qué tan bien h se aproxima a f, se define una medida o función de error, que cuantifique la proximidad de $\hat{y}$ respecto al valor esperado y [79]. Los algoritmos de ML supervisados aprenden a través de un proceso de optimización tratando de encontrar aquellos valores de los parámetros tal que minimicen el error total del modelo.

Para medir la precisión de una hipótesis, se le da un conjunto de prueba con ejemplos distintos al conjunto de entrenamiento. De esta forma, se dice que una hipótesis se generaliza bien si predice correctamente de una manera "razonable" el valor de y para ejemplos nuevos [81].

2.2.3 Proceso

Cross Industry Standard Process for Data Mining (CRISP-DM) es el modelo estándar abierto de proceso de facto [82] mayormente utilizado en proyectos de minería de datos y descubrimiento de conocimiento [83]. El proceso, independientemente de la industria y tecnología utilizada, proporciona una descripción general en seis fases iterativas (ver Figura 2.7) del ciclo de vida de un proyecto de ciencia de datos o ML, ayudando a su planificación, organización e implementación [84]:

- *Comprensión del negocio*. Etapa sumamente esencial del ciclo de vida que sienta las bases del proyecto donde se investigan los objetivos y requerimientos del mismo. Es fundamental una comprensión clara del problema a resolver, su impacto y objetivos a fin de determinar la factibilidad del proyecto.

- *Comprensión de los datos*. Se identifica, recopila, analiza y controla la calidad del conjunto de datos disponible a fin de determinar su factibilidad para futuro procesamiento. Sin datos no es posible continuar con un proceso de ML, siendo además muy importante que sean de calidad y variedad.

- *Preparación de los datos*. Se realizan un conjunto de tareas destinadas a transformar el conjunto de datos de forma tal que los algoritmos de ML puedan producir modelos. Es una etapa crucial ya que los datos son la piedra angular del proceso de ML, malos datos conducirán a la generación de modelos con rendimiento y resultados deficientes [85].
- *Modelado*. Etapa central basada en la construcción iterativa de múltiples modelos a fin de encontrar el o los mejores que satisfagan los requerimientos y objetivos deseados.
- *Evaluación*. Se realiza una valoración y revisión más abarcativa de los modelos finales junto con sus resultados a fin de determinar el que mejor se adapta a las necesidades.
- *Despliegue*. Se integran los modelos seleccionados productivamente para el acceso a sus resultados. Luego se mantiene una monitorización constante de los mismos.

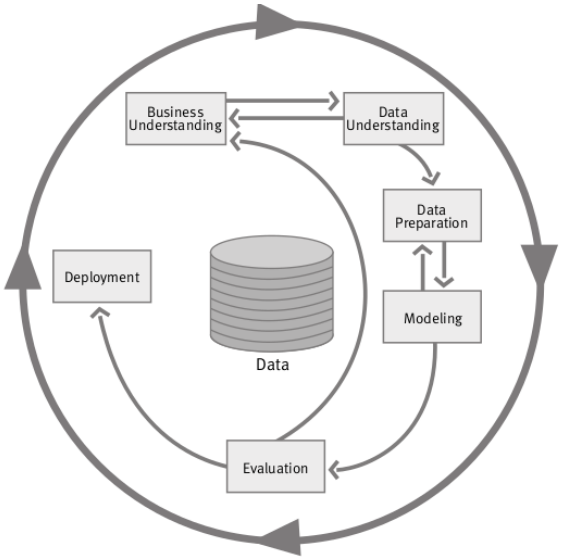


Figura 2.7: Fases del modelo de referencia CRISP-DM. Extraído de [84].

Posiblemente, la parte más importante del proceso de ML es comprender los datos con los que está trabajando y cómo se relacionan con la tarea a resolver. Sin las dos primeras etapas no es posible avanzar con un proceso de ML, no siendo efectivo elegir aleatoriamente un algoritmo y entregarle datos. Cada algoritmo de ML es diferente en términos de qué tipo de datos y para qué configuración de problemas funciona mejor [66]. Asimismo, la etapa de preparación es vital a tal punto que en general se destina un 80% del tiempo total del ciclo de vida del proyecto [85]. Ningún algoritmo de ML podrá hacer una predicción sobre datos para los que no tiene información [66]. Con esto es posible deducir que se requiere una mínima intervención humana e intuición para llevar adelante un proceso de ML.

En un contexto más amplio, los algoritmos y métodos en ML son solo una parte de un proceso mayor para resolver un problema en particular, y es bueno tener en cuenta el panorama general en todo momento [66]. A partir del modelo CRISP-DM presentado, es posible identificar ciertas etapas principales para la presente investigación relacionadas con la preparación de los datos (2.2.3.1), modelado (2.2.4) y evaluación (2.2.5).

2.2.3.1 Preparación de datos

La explosión del siglo XXI radicada en el uso de las computadoras, internet, dispositivos móviles, sensores y demás, produjo que los datos sean el centro de todo lo que nos rodea [86]. Diariamente, numerosas fuentes alrededor del mundo generan datos a diferentes velocidades en numerosos formatos, formas y tamaños [66]. Asimismo, cada conjunto de datos estará compuesto por ejemplos con *features*, que generalmente, se agrupan en los siguientes tipos:

- *Numérico*. Tipo más simple que representa información escalar sobre las entidades observables, como por ejemplo el peso de una persona.
- *Texto*. Tipo más común que comprende contenido alfanumérico no estructurado el cual requiere un esfuerzo adicional para su transformación y comprensión.
- *Categórico*. Tipo referido a categorías de entidades que se observan, como por ejemplo el nivel económico bajo, medio o alto. Los valores pueden representarse como numéricos o alfanuméricos.

El proceso de preparación de los datos involucra un conjunto de subprocesos destinados a limpiar, transformar y mapear los datos crudos a fin de que sean utilizables para su análisis, visualización, sumarización, reportes, y, por sobre todo, los algoritmos de ML. Este proceso es muy importante y complejo afectando directamente a las etapas posteriores del mismo.

Exploración y visualización de datos Es muy importante realizar un análisis descriptivo y exploratorio inicial del conjunto de datos disponible. La comprensión y descripción del conjunto de datos permite tener un mejor entendimiento del problema así como también comprender las limitaciones dadas por los *features* disponibles. Explorar la cantidad de ejemplos y *features*, sus nombres y tipos de datos, la presencia de valores faltantes o nulos, son algunas tareas que permitirán dar el puntapié inicial para la comprensión del conjunto de datos.

La visualización de datos refiere al proceso de representar visualmente la información mediante tablas, gráficos, imágenes, mapas, etc, a fin de poder tener una noción gráfica de su distribución. Esta es una herramienta de mucho valor que permite no sólo una mejor comprensión de los datos sino también una detección y corrección temprana de errores en los mismos. Sin su aplicación, estos errores podrían propagarse a etapas posteriores generando resultados indeseados o inconsistencias difíciles de identificar.

La Matriz de Correlación (MC) es una forma de visualización donde se muestran los coeficientes de correlación entre conjuntos de variables [87]. Cada punto de la tabla muestra la correlación entre dos variables y su representación permite resumir una gran cantidad de datos con el objetivo de identificar patrones (por ejemplo, de variables fuertemente correlacionadas). El coeficiente de correlación es una medida de la fuerza y dirección en la relación lineal entre dos variables, que van desde -1 (correlación negativa perfecta) a 1 (correlación positiva perfecta). Un valor de 0 indica que no hay correlación entre las variables.

Preprocesamiento de datos Dentro de esta etapa se abren un conjunto de actividades relacionadas con la limpieza y transformación de los datos en información adecuada para su posterior procesamiento. Es importante remarcar que existe una gran cantidad de tareas de las cuales quizás no todas deben ser aplicadas, ya que depende de cada problema y conjunto de datos particular.

Manejo de valores faltantes o nulos Los datos faltantes se definen como los valores que no están almacenados (o no están presentes) para alguna variable en el conjunto de datos dado. Generalmente, en un conjunto de datos real se tienen valores faltantes o nulos producto de distintos factores [88], [89], los cuales pueden generar diversos problemas [90]. En particular, suelen conducir a una interpretación incorrecta por los algoritmos de ML generando errores en los cálculos y/o resultados finales. Además, la gravedad de los valores faltantes depende en parte de la cantidad de datos omitidos, el patrón y el mecanismo subyacente a la falta de datos [90]. Por tal motivo, es importante lidiar con estos antes de analizar los datos, ya que ignorar u omitir los valores faltantes puede resultar en un análisis sesgado o mal informado [91].

El primer paso para manejar los valores faltantes es observar los datos cuidadosamente, analizando la cantidad de valores nulos para cada variable a fin de comprender las razones que ayuden a determinar la mejor estrategia para su manejo [92]. Existen dos técnicas principales para su manejo:

- *Eliminación*. En este enfoque, más simple, todas las entradas con valores nulos se eliminan. Sin embargo, se ha demostrado que la eliminación puede introducir sesgos en los datos, especialmente cuando aquellos que faltan no se distribuyen aleatoriamente [93]. Puede suponerse que este enfoque es aconsejable sólo cuando la proporción de datos a eliminar es pequeña. El proceso de borrado puede llevarse a cabo por [94]:
  - *Eliminación de registros completos*. Se eliminan todos los registros en los que falta algún valor, lo que lleva a disminuir el tamaño de la muestra. Si la cantidad de datos es suficientemente grande y la ausencia se distribuye aleatoriamente entonces este enfoque puede ser razonable. De lo contrario, este enfoque puede resultar en la pérdida de información importante, especialmente cuando la cantidad de casos descartados es considerable.

  - *Eliminación por columna*. Si un *feature* contiene muchos valores faltantes, por ejemplo más del 80%, y no es significativo, eliminarlo puede ser una opción viable. A diferencia del caso anterior, esta opción puede ser conveniente si el *feature* cuenta con muchos valores nulos. De esta manera, se prioriza mantener el tamaño de la muestra a costa de eliminar quizás algún *feature* con pocos datos nulos.

- *Reemplazo*. Este proceso implica reemplazar los valores faltantes por valores sustituidos. En general, aquellos valores no nulos se utilizan para predecir los valores a sustituir [95]. Dependiendo de la naturaleza del problema, las técnicas de reemplazo pueden ser ampliamente clasificadas de la siguiente manera:

  - *Reemplazo simple*. Se reemplazan los valores faltantes mediante el uso de un atributo cuantitativo o cualitativo de todos los valores no nulos [96].
  - *Reemplazo basada en ML*. Estas son técnicas sofisticadas que en su mayoría implican el desarrollo de un enfoque predictivo para manejar los valores faltantes mediante el aprendizaje supervisado o no supervisado.

Aunque existan muchos enfoques, es importante señalar que la única solución adecuada se reduce a un buen análisis de los datos y comprensión del dominio que permita tomar la mejor decisión. De lo contrario, un manejo inadecuado puede conducir a inferencias inexactas [97].

Transformaciones Los algoritmos de ML experimentados en la presente investigación se desempeñan principalmente con datos numéricos, mientras que las variables categóricas pueden plantear diversos problemas. Por tal motivo, es importante realizar una transformación de estas variables a una representación numérica. Asimismo, en algunos casos las estructuras de datos creadas para contener al conjunto de datos deben ser modificadas a fin de facilitar ciertos pasos intermedios del procesamiento del modelo. Nuevos *features*, o incluso nuevas estructuras transitorias pueden ser creadas para facilitar cálculos auxiliares, identificar determinadas observaciones y cruzar dos o más estructuras de datos.

Normalización La normalización es el proceso de estandarizar el rango de valores de los *features* o características. En la mayoría de las ocasiones, los conjuntos de datos cuentan con valores numéricos de naturaleza completamente diferente. El uso de los valores como *features* de entrada tal y como están puede ocasionar que los modelos sesguen hacia aquellos que tengan valores de magnitud realmente altos. Aunque existen modelos sensibles a la escala, como la Regresión Lineal o Regresión Logística, y otros que no, como los métodos basados en árboles, se recomienda realizar una normalización especialmente si se desea probar varios algoritmos de ML sobre ellos. Las principales formas de normalización son:

- *Normalización min-max*. Se estandarizan los valores de cada *feature* de forma tal que dicho valor esté dentro del rango [0-1]. En términos matemáticos sería:

$$MinMax(X_i) = \frac{X_i - min(X)}{max(X) - min(X)}$$

  Donde a cada valor en el *feature* X_i se le resta el mínimo de dicho *feature* $min(X)$ y normaliza por la diferencia entre el máximo $max(X)$ y el mínimo $min(X)$.

- *Normalización estándar*. A este proceso también se lo conoce como normalización Z-score, y puede ser denotado matemáticamente como:

$$Z - score(X_i) = \frac{X_i - \mu_x}{\sigma_x}$$

  Donde cada valor en el *feature* X_i se resta por la media de dicho *feature* μ_x y se normaliza por la desviación estándar σ_x. Entonces, luego de la resta se tiene la media de cada distribución en 0 y luego de normalizar por la desviación estándar se tiene la desviación en 1 para todos los *features*. Esto no significa que exista un máximo y un mínimo.

Limpieza La limpieza de un conjunto de datos implica tareas como la eliminación y gestión de datos incorrectos o faltantes, el manejo de valores atípicos[7] (*outliers*), etc. Asimismo, también implica estandarizar los nombres de los atributos para que sean más legibles e intuitivos.

Específicamente, los *outliers* pueden tener un impacto negativo sobre los modelos entrenados con el conjunto de datos al cual pertenecen [98]. Por eso, es importante realizar un estudio a fin de identificarlos para posteriormente tratarlos. Aunque existen diversas técnicas abocadas a estos objetivos, en la presente investigación se aplico el análisis de rango intercuartil, también conocido como método de Tukey, junto con una técnica de visualización conocida como diagramas de caja (*boxplot*) [87], [99].

- *Diagrama de caja (boxplot)*. Representación gráfica que muestra la distribución de los datos a través de sus cuartiles [100]. La Figura 2.8 muestra sus elementos principales:

[7] Observación o punto de datos que difiere significativamente de otras observaciones en el conjunto de datos. Los valores atípicos pueden ser el resultado de errores de medición, errores en la entrada de datos, variación natural de los datos, o indicar una ocurrencia legítima aunque rara en la población estudiada.

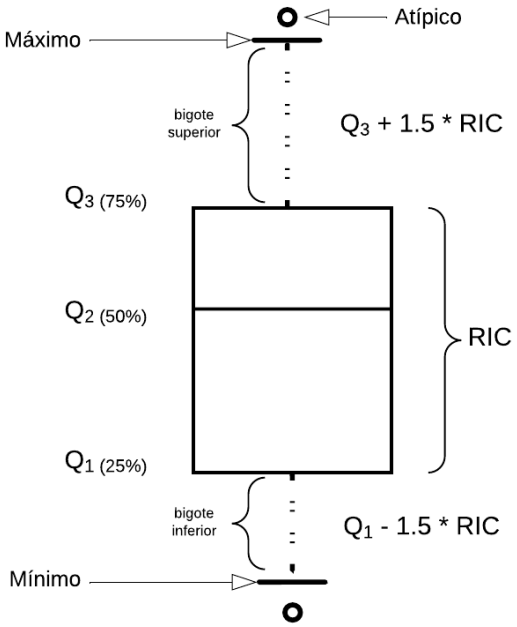


Figura 2.8: Componentes principales de un diagrama de caja.

- *Caja o rectángulo*. La caja o rectángulo representa el Rango Intercuartil (RIC), que es la distancia entre el primer cuartil (Q_1)[8] y el tercer cuartil (Q_3)[9], o bien, $RIC = Q_3 - Q_1$. El extremo inferior de la caja representa el Q_1 y el extremo superior del rectángulo representa el Q_3.
- *Mediana*. La mediana (Q_2) es el valor medio en un conjunto de datos ordenados en orden ascendente o descendente[10].
- *Bigotes*. Los bigotes representan el rango de los datos que quedan fuera de la caja. La longitud de los bigotes se puede determinar de diferentes maneras, una de ellas es utilizar la regla empírica de 1.5 veces el RIC, así: $bigote\_inferior = Q_1 - 1.5 * RIC$ y $bigote\_superior = Q_3 + 1.5 * RIC$.
- *Valor mínimo y máximo*. Estos valores se representan como los extremos de los bigotes.
- *Valores atípicos*. Aquellos puntos de datos que se encuentran fuera de los bigotes se consideran *outliers* y, por lo general, se representan como puntos individuales o asteriscos en el gráfico.

[8] Representa el 25% de los datos.

[9] Representa el 75% de los datos.

[10] No se debe confundir este término con la media, la cual representa el promedio de un conjunto de datos. La moda es el valor que aparece con más frecuencia en un conjunto de datos.

- *Análisis de rango intercuartil*. También conocido como método de Tukey, es una técnica estadística basada en el cálculo y utilización del valor RIC para identificar los *outliers* en un conjunto de datos. Luego, se definen los siguientes límites: $Li = Q_1 - k * RIC$ y $Ls = Q_3 + k * RIC$ donde $k = 1.5$ es utilizado para identificar *outliers* leves y $k = 3$ para *outliers* extremos. Así, aquellos puntos de datos que pertenezcan a alguno de los intervalos $[Q_1 - 3 * RIC; Q_1 - 1.5 * RIC)$ o $(Q_3 + 1.5 * RIC; Q_3 + 3 * RIC]$ serán considerados *outliers* leves. Por otra parte, aquellos puntos de datos inferiores a $Q_1 - 3 * RIC$ o superiores a $Q_3 + 3 * RIC$ serán considerados *outliers* extremos [99].

Resulta importante destacar que la detección de *outliers* debe ser cuidadosa y considerada en el contexto del problema específico. Algunos valores que podrían parecer *outliers* en un análisis, pueden ser explicados por factores específicos del problema o pueden ser valores extremos legítimos.

Reducción de dimensionalidad Trabajar con conjuntos de datos que se encuentran sobrepasados de *features* puede resultar inconveniente, ya que esto genera un espacio de gran dimensión y de complejidad alta. En este contexto, los problemas que se presentan se relacionan al análisis y visualización de los datos, tiempo y memoria requerida para el entrenamiento de los modelos, limitaciones de espacio y generación de modelos complejos propensos a un sobreajuste (concepto abordado en la Sección 2.2.5) y de difícil interpretación [85]. La reducción de la dimensionalidad es el proceso de reducir el número original de *features* a un subconjunto significativo, idealmente cerca de su dimensión intrínseca[11] [102]. Las técnicas de reducción de dimensionalidad pueden ser clasificadas en dos enfoques principales: Selección de *features* y Extracción de *features*.

Selección de features La premisa principal cuando se utiliza este enfoque es que, en general, los conjunto de datos contienen *features* redundantes o irrelevantes[12] y, por lo tanto, se pueden eliminar sin generar pérdida de información [103]. Un algoritmo de selección de *features* puede verse como la combinación de una técnica de búsqueda para encontrar nuevos subconjuntos de *features*, junto con una medida de evaluación que puntúa los diferentes subconjuntos. Quizás el algoritmo más simple consiste en realizar pruebas desde todo el conjunto *features* hasta el mínimo subconjunto a fin de encontrar aquel que minimice la tasa de error. Esta es una búsqueda exhaustiva del espacio, que resulta computacionalmente compleja en aquellos conjuntos de datos con gran cantidad de *features*.

Extracción de features Busca extraer, derivar o transformar información del conjunto de datos original a fin de crear un nuevo subespacio de *features* destinados a

[11] La dimensionalidad intrínseca de un espacio es el número de piezas de información requeridas para describir cada objeto en el espacio. Esto puede diferir de la cantidad de piezas de información utilizadas, llamada la dimensionalidad extrínseca del espacio [101].

[12] Redundante e irrelevante son dos conceptos distintos, un *feature* relevante puede ser redundante en presencia de otra característica relevante con la que está fuertemente correlacionada.

ser informativos[13] y no redundantes bajo la premisa de conservar la información [104]. En general, el proceso de extracción de *features* también tiene su aplicación para la visualización del espacio en una dimensión más reducida, la cual facilite la comprensión del problema y su complejidad asociada. A continuación se presentan los algoritmos no supervisados utilizados en la presente tesina para dicha transformación:

- El Análisis de Componentes Principales (PCA, por sus siglas en inglés) es un método estadístico que permite simplificar la complejidad de un conjunto de *features* de mayor dimensión, posiblemente correlacionado, en un conjunto de menor dimensión de *features* linealmente no correlacionados conservando a la vez su información. Estos *features* transformados también se conocen como Componentes Principales (PC, por sus siglas en inglés). En cualquier transformación de PCA, la cantidad total de PC siempre es menor o igual que la cantidad inicial de *features*. El primer PC trata de capturar la varianza máxima del conjunto original de *features*. Cada uno de los componentes subsiguientes trata de capturar las siguientes varianzas máximas de modo tal que sean ortogonales a los componentes anteriores asegurando así la no correlación. De esta forma, existirán tantos PC como cantidad de *features*.

- Por su lado, la Incrustación de Vecinos Estocásticos Distribuidos en t (t-SNE, por sus siglas en inglés) es un método estadístico utilizado principalmente para la visualización de datos de alta dimensionalidad dando a cada punto de datos una ubicación en un espacio tridimensional o bidimensional de manera tal que las relaciones entre los puntos sean preservadas tanto como sea posible. Su característica principal radica en ser una técnica de reducción de dimensionalidad no lineal (a diferencia de PCA) permitiendo tratar con datos linealmente no separables. La técnica construye una distribución de probabilidad para cada punto en el espacio de alta dimensión, que refleja la probabilidad de que otros puntos estén cerca de él. Luego, se construye otra distribución de probabilidad para cada punto en el espacio de baja dimensión, y se ajustan los parámetros de la distribución de manera que la distribución de probabilidad de los puntos en la baja dimensión sea similar a la distribución de probabilidad en la alta dimensión.

Por otra parte, en la presente tesina, también se utilizaron los siguientes algoritmos supervisados de reducción de dimensionalidad:

- El Análisis de Componentes de Vecindad [105] (NCA, por sus siglas en inglés) es un algoritmo de aprendizaje de la distancia Mahalanobis a ser utilizado para maximizar la *performance* del modelo de clasificación K vecinos más cercanos. Sutilmente, este algoritmo traduce dicho aprendizaje a la búsqueda de una matriz de proyección (o transformación) lineal a través de una optimización del error de clasificación de exclusión esperado en el espacio transformado. De esta manera, la métrica de distancia aprendida o, de manera equivalente, la matriz de proyección

[13] Los *features* informativos son aquellos que brindan información para la construcción del modelo, no son redundantes y su agregación al entrenamiento produce mejoras.

se encuentra directamente relacionada con el rendimiento de clasificación. Al restringir la matriz de proyección a una no cuadrática, NCA puede utilizarse para la reducción de dimensionalidad [105], [106].

- Por su parte, el Análisis Discriminante Lineal (LDA, por sus siglas en inglés), es una generalización del discriminante lineal Fisher's [107], similar a PCA algorítmicamente, aunque, en lugar de maximizar la varianza, LDA se centra especialmente en identificar aquellos atributos que maximicen la separabilidad entre las clases conocidas [108]. A fin de realizar esto, LDA utiliza la información de los *features* con el objetivo de trazar el nuevo espacio reducido y, así, proyectar en este el conjunto de datos de forma tal de maximizar la separación entre las clases dadas. Con el objetivo de no ser redundante, vale la pena mencionar la única diferencia de LDA respecto a PCA radicada en el cálculo de la máxima separabilidad entre clases. Mientras que PCA busca maximizar la varianza de los datos sin considerar las etiquetas de clase, LDA considera las etiquetas de clase y busca maximizar la separación entre estas y minimizar la variación de los datos dentro de las clases.

2.2.4 Modelado

A continuación se describen algunos de los modelos predictivos más populares, los cuales fueron utilizados en la presente investigación.

2.2.4.1 Regresión Lineal

El modelo de Regresión Lineal Simple (SLR, por sus siglas en inglés), es un modelo lineal de los más simples y conocidos para la resolución de problemas de regresión. Este modelo asume que existe una relación lineal entre una variable independiente x y otra dependiente y, desarrollándose una predicción aproximada dada por la función lineal recta: $y = mx + b$ o bien visto como una función que representa el modelo $f(x) = mx + b$. De esta forma, el objetivo será predecir para nuevos valores de x, el valor de y asociado.

En dicha función, tanto m como b, la pendiente y ordenada al origen respectivamente, definen la recta y representan los parámetros del modelo. Una combinación particular de estos, representarán un caso particular dentro de la familia de modelos de SLR, donde para cada valor particular del parámetro se tendrá un modelo distinto dentro de la misma.

Función de error En general, en función de la familia de modelos, será necesario un criterio objetivo para la selección de los mejores parámetros en función del problema a resolver. Aquí se introduce el concepto de función de error $E(y', y)$, cuyo objetivo será medir el error de cada modelo, dado por la diferencia entre los valores predichos y' y esperados y, a fin de determinar el de mejor proximidad o, en otras palabras, qué tan certeras son las predicciones de cada uno. Entonces, dado un conjunto de prueba (x, y), de tamaño N es posible calcular el error total E de un modelo f en función de sus parámetros θ: $E(\theta) = \frac{1}{N} \sum_i^N E(f(x_i), y_i)$.

La función de error natural en SLR se encuentra definida por el Error Cuadrático Medio (MSE, por sus siglas en inglés) definido como: $MSE(m,b) = \frac{1}{N}\sum_i^N E_i(m,b)$ donde $E_i(m,b)$ representa el error de un punto i para los parámetros del modelo. En general, $E_i(m,b)$ para SLR se define como el error cuadrático dado por la distancia euclidiana cuadrática entre el valor esperado y_i y el valor predicho $f(x_i)$: $E_i(m,b) = (y_i - f(x_i))^2 = (y_i - mx_i + b)^2$. De esta manera, el problema se resume en encontrar aquella combinación óptima de los parámetros del modelo resultante del proceso de minimización de la función de error.

Optimización de funciones - descenso de gradiente La búsqueda de los parámetros óptimos en los modelos de aprendizaje supervisado se realiza en base al conjunto de entrenamiento conformado por un par de vectores x e y de tamaño N, que representan ejemplos de entrenamiento x_i y su correspondiente valor esperado respectivamente y_i. De esta forma, el objetivo será aprender los parámetros óptimos en base a los datos. Este proceso se denomina aprendizaje o entrenamiento, y, en la mayoría de los casos, se desencadena en un proceso de optimización de la función de error, la cual refiere a la búsqueda de alguno de sus mínimos[14] mediante la variación de sus parámetros.

Uno de los métodos comúnmente utilizados para la optimización de funciones es el descenso de gradiente [62], [109], el cual consiste de un algoritmo iterativo que asume que la función a optimizar es derivable utilizando el gradiente o derivada como medio para guiar la optimización de la misma. El algoritmo básico comienza con un valor aleatorio del parámetro e itera hasta el criterio de convergencia. En cada iteración, se calcula el gradiente en un punto para continuar hacia la dirección opuesta. Este algoritmo es muy importante ya que es generalizable a la mayoría de los modelos que se abordan en la presente investigación.

Dada una función f con sus respectivos parámetros w_i, una función de error E, y un conjunto de entrenamiento compuesto por los vectores x e y de entrada y su correspondiente valor, el algoritmo básico calcula iterativamente el vector gradiente de la función de error E respecto a los parámetros w_i habiéndose fijado los valores del conjunto de entrenamiento. En cada iteración, se actualizan los valores de w_i sustrayendo el valor del vector gradiente calculado en el punto w_i multiplicado por un valor α conocido como *learning rate* o factor de aprendizaje, lo que equivale a desplazarse sobre E en la dirección de decrecimiento en ese punto. En la Figura 2.9, puede visualizarse el proceso de actualización del algoritmo, el cual se repite hasta alcanzar el criterio de convergencia, es decir, alcanzar un valor mínimo. Llegado ese punto, no existirá variación en el valor de la función a optimizar.

[14] Dependiendo de la función de error del modelo podrá existir un único mínimo global (función convexa) o bien varios mínimos (función no convexa).

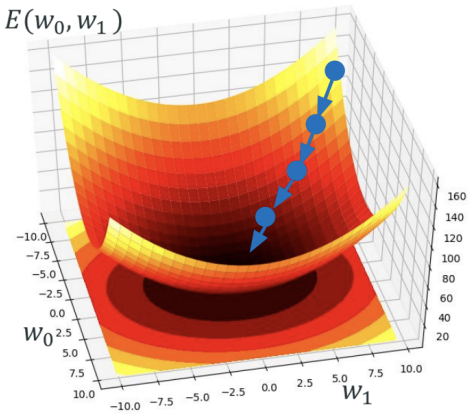


Figura 2.9: Proceso de descenso de gradiente en un espacio tridimensional con una función de error a optimizar de dos parámetros $E(w_0, w_1)$.

Es importante mencionar que α se considera un hiperparámetro por no encontrarse dentro de los verdaderos parámetros del modelo a entrenar. Este hiperparámetro indicará la velocidad de aprendizaje o tamaño de los saltos sobre la dirección de desplazamiento del algoritmo en cada iteración. El tamaño de α es sumamente importante ya que:

- Si es muy pequeño, se avanzará poco en cada iteración desencadenando en un alto costo computacional.
- Si es demasiado grande, se tendrán desplazamientos muy grandes pudiendo ocurrir que el algoritmo no converja.
- Si es balanceado o correcto, se tendrá un buen avance por iteración con un costo computacional razonable.

Si bien el criterio de convergencia general será alcanzar un valor mínimo, esto puede conducir al problema de haber llegado a uno sin que éste sea un óptimo. Estos son llamados puntos silla. También puede ocurrir que el mínimo alcanzado sea un mínimo local, es decir, que es mínimo en cierta región. En general, varias de las funciones de error utilizadas en el contexto de las Redes Neuronales Artificiales pueden tener varios mínimos locales, lo que hace más difícil la búsqueda. En estos casos, alcanza con encontrar un punto cuyo error sea lo suficientemente bajo.

Aunque se hizo mención al modelo SLR, éste representa una base para las variaciones del mismo como la Regresión Lineal (LR, por sus siglas en inglés). Así como el modelo SLR se encuentra representado por una única variable de entrada ($x \in R^1$), la extensión del mismo refiere al modelo LR donde se tienen múltiples variables de entrada ($x \in R^m$), teniendo más información de los ejemplos distribuidos en un espacio m-dimensional. La función que representa el modelo LR resulta de una generalización de SLR: $f(x) = x \cdot w + b$, donde para m coeficientes o pesos se tendrá que x y w son vectores de la forma $(x_1, x_2, ..., x_m)(w_1, w2, ..., w_m)$ respectivamente. Debido a esto, la función de error y la aplicación del descenso de gradiente para su optimización tampoco se ven modificadas, sólo que ahora serán $m + 1$ parámetros a optimizar.

2.2.4.2 Regresión Logística

La clasificación binaria o en dos clases es un problema de clasificación dirigido a determinar para una entrada su pertenencia a una de dos clases correspondientes, las cuales son típicamente representadas por el conjunto $\{0,1\}$ y también llamadas clase negativa y positiva, respectivamente. Este tipo de problemas y la predicción de probabilidades se encuentran íntimamente relacionados con el poder convertir unidades de probabilidad en clases mediante el umbral de detección definido por un valor entre 0 y 1 que especifica el corte sobre el cual se transformarán las probabilidades en clases.

La relación anteriormente descrita hace que aunque ambos problemas sean distintos, se encuentren relacionados bajo el mismo modelo, el cual será entrenado con datos de clasificación $\{0,1\}$ a fin de realizar predicciones de probabilidad. Luego, aplicando el umbral, se podrá asociar cada una a su correspondiente clase binaria.

El modelo de Regresión Logística (LOR, por sus siglas en inglés[15]) es uno de los más simples y a la vez importantes[16] para realizar tanto clasificación binaria como predicción de probabilidades. Una posible forma de obtener este modelo es a través de la combinación del modelo de LR con la función sigmoidea σ definida como: $\sigma(x) = \frac{1}{1+e^{-x}}$, donde su dominio se encuentra en el intervalo $[-\infty, +\infty]$ y su imagen se encuentra en el intervalo $[0,1]$, como puede visualizarse en la Figura 2.10. La combinación permite restringir o transformar la salida del modelo LR a un valor en el dominio $[0,1]$ permitiendo ser interpretado como una probabilidad. De esta forma, se define el modelo LOR como: $f(x) = \sigma(mx+b) = \frac{1}{1+e^{-mx-b}}$ o, en su forma general con múltiples variables de entrada, como: $f(x_1, x_2, ..., x_n) = \sigma(x_1 w_1, x_2 w_2, ..., x_n w_n + b)$.

[15] La referencia como LOR se realiza para su distinción respecto a LR.

[16] La importancia radica en su utilización como base dentro de las Redes Neuronales que realizan clasificación.

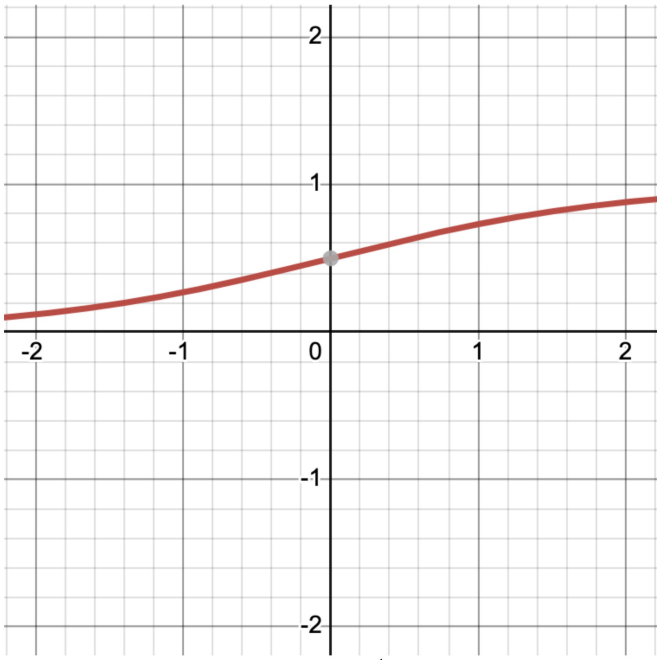


Figura 2.10: Gráfica de la función sigmoidea con su típica forma de "S".

A partir del modelo LOR, cuya salida es una probabilidad, será posible transformar mediante la utilización de un umbral cada valor a su correspondiente clase $\{0,1\}$. El resultado definirá implícitamente un límite o frontera de decisión, la cual separará ambas clases correspondientes en el plano N-dimensional.

Debido a que LOR cambia la función de decisión respecto a LR agregando una función sigmoidea, también la función de error se ve modificada. Para el modelo LOR de clasificación binaria la función de error utilizada es la Entropía Cruzada Binaria (BCE, por sus siglas en inglés) definida como: $E(\theta) = \frac{1}{N} E_i(f(x), y)$, donde $f(x)$ será el valor predicho mientras que y el verdadero, en tanto $E_i(f(x), y) = y_i[-log(f(x_i))] + (1 - y_i)[-log(1 - f(x_i))]$ siendo $f(x_i) = \sigma(mx_i + b)$.

Si bien queda por fuera de la presente investigación adentrarse sobre la formulación matemática de las distancias asociadas, vale la pena mencionar que así como MSE mide distancias entre la predicción del modelo y el valor verdadero utilizando la distancia euclídea al cuadrado para determinar el error, BCE mide la distancia entre distribuciones de probabilidad entre lo predicho y lo esperado basándose en la distancia de Kullback-Leiber [110].

Pese a que las funciones de error para LR y LOR son distintas, la forma de optimización mediante el descenso de gradiente es la misma ya que sus derivadas son de la misma forma, lo que vislumbra el poder de generalización para dicho algoritmo.

2.2.4.3 Árbol de Decisión

El Árbol de Decisión (DT, por sus siglas en inglés) es un modelo ampliamente utilizado para problemas de clasificación y regresión [66]. Esencialmente, este aprende una jerarquía de preguntas "if-else", que conducen a una decisión, inferidas a partir del conjunto de *features*. El objetivo radica en llegar a la predicción de la pertenencia a una clase o un valor continuo bajo la mínima altura del árbol posible, facilitando su comprensión [66]. A continuación se mencionan algunas definiciones importantes:

- La parte superior del árbol se denomina nodo raíz.
- Los nodos internos o de decisión son aquellos que contienen las condiciones para la bifurcación de los datos. El nodo raíz también es un nodo interno.
- Las hojas son aquellos nodos finales que ayudan a predecir la clase asociada o el valor continuo asociado al nuevo punto de datos.

Un DT de clasificación es, en forma genérica, un árbol binario que recursivamente divide el conjunto de datos realizando una clasificación a través de un conjunto de decisiones para cada *feature*, comenzando desde el nodo raíz del árbol y progresando hasta las hojas en función de las decisiones, donde se recibe la decisión de una clase particular a partir de los datos contenidos en la misma.

Aunque existen diferentes algoritmos para la implementación del DT, casi todos son variantes del mismo principio: los algoritmos construyen el árbol utilizando una heurística voraz durante su entrenamiento, comenzando desde la raíz, evaluando las posibles opciones en la etapa actual de aprendizaje y seleccionando aquel *feature* más informativo en ese punto. Aunque si bien por un lado no se garantiza la selección óptima global del conjunto de decisiones, el entrenamiento del modelo será más rápido respecto a aquellos que realizan una vuelta hacia atrás.

La selección del *feature* a utilizar en cada paso se basa en una medida cuantitativa que representa la mayor cantidad de información, matemáticamente esto se encuadra dentro de la teoría de la información. En [111] se propuso la medida de la Entropía de la Información, la cual describe la cantidad de impureza en un conjunto de *features*. La entropía H de un conjunto de probabilidades p_i es: $H(p) = \sum -p_i log(p_i)$, donde p_i es la probabilidad de la clase i, y, cuanto más alta es la medida (próxima a 1), se tendrá una mayor incertidumbre sobre la clase del nuevo punto de datos. Una forma de aplicación de esta medida es el cálculo de cuánto disminuiría la entropía de todo el conjunto de entrenamiento si se elige cada *feature* posible para el siguiente paso de clasificación. Esto se conoce como la Ganancia de Información (IG, por sus siglas en inglés), y se define como la entropía de todo el conjunto menos la entropía cuando se elige un *feature* en particular: $IG(S,F) = H(S) - \sum_{f \in values(F)} \frac{|S_f|}{|S|} H(S_f)$, donde S es el conjunto de ejemplos, F es un *feature* de todos los posibles, y $|Sf|$ es la cantidad total de miembros de S que tienen el valor f para el *feature* F.

La construcción de un DT de regresión no requiere grandes cambios respecto a su implementación. A fin de realizar una predicción, se recorre el árbol según las condiciones

hasta llegar a las hojas donde se establecen valores que representan el promedio del conjunto de datos de entrenamiento asociados a la misma. Debido a que las predicciones son valores numéricos, se utilizará MSE como medida de error a minimizar en cada paso para la selección de un *feature*.

2.2.4.4 Random Forest

A diferencia de los métodos típicos, que producen un único modelo, el objetivo de los métodos de ensamblado radica en combinar las predicciones de varios modelos base, de acuerdo a un criterio, a fin de mejorar la generalización o robustez en comparación con un solo estimador [112]. La combinación de muchos estimadores ligeramente diferentes generarán resultados significativamente mejores que cualquiera de ellos por separado, dándose tanto para conjuntos de entrenamiento grandes como pequeños (siempre que dicha combinación sea realizada correctamente) [80]. Existen dos familias de métodos de ensamble:

- *Bagging*. Proveniente del concepto de *bootstrap aggregation* [113], el objetivo fundamental es construir varios modelos débiles de manera independiente, para luego promediar sus estimaciones. En promedio, el modelo ensamblado suele ser mejor que cualquiera de los modelos base por simplificar la solución y reducir la varianza evitando el sobreajuste. Un ejemplo dentro de esta familia está dado por el algoritmo *Random Forest* (RF) abordado a continuación.

- *Boosting*. Los modelos base se construyen secuencialmente con el objetivo de reducir el sesgo del estimador ensamblado. La motivación de esta familia es combinar varios modelos débiles a fin de producir uno robusto [80]. Algunos ejemplos dentro de esta familia son los algoritmos *XGBoost* o *AdaBoost*, los cuales pueden profundizarse en [80], [114], [115].

Una de las principales desventajas de los DT está dada por su tendencia hacia el sobreajuste, en ese sentido RF busca abordar este problema. La idea bajo el modelo RF, que ha ganado popularidad en los últimos años [80], radica en pensar que si un DT puede realizar una predicción relativamente buena entonces muchos DT (un bosque) deberían ser mejores, siempre que exista suficiente variabilidad entre ellos. Esto permite que cada DT se sobreajuste de distintas formas, logrando así reducir dicho efecto al promediar sus predicciones. Esto significa que el modelo RF representa una colección de DT, donde cada uno es ligeramente diferente a los demás [66].

Derivado de su nombre, RF realiza una inyección de aleatoriedad sobre cada DT construido para asegurar su variabilidad: seleccionando el conjunto de entrenamiento para construir cada DT y el subconjunto de *features* en cada paso del algoritmo [66].

Debido a que en la práctica no suele disponerse más que una única muestra del conjunto de datos, el algoritmo toma la llamada muestra *bootstrap*: a partir del conjunto de datos de tamaño N, se extrae repetidamente un ejemplo al azar con reemplazo (lo que significa que la misma muestra se puede seleccionar varias veces) N veces, creando

así un conjunto de datos igual en cantidad al original con ausencia de algunos ejemplos (aproximadamente un tercio) aunque repitiendo alguno de ellos.

A partir de cada conjunto de datos creado, se crea cada DT de forma independiente con una modificación en su algoritmo: en cada nodo, el algoritmo selecciona aleatoriamente un subconjunto de *features* y busca el mejor posible que involucre uno de ellos. Este procedimiento se repite por separado en cada nodo, de modo que cada uno en un árbol pueda tomar una decisión utilizando un subconjunto diferente de *features*.

De esta forma, mediante los mecanismos de *bootstrapping* y selección de *features* en cada nodo, se asegura aleatoriedad entre los árboles del bosque. Es importante remarcar que la cantidad máxima de *features* a considerar y la cantidad de DT a generar son hiperparámetros posiblemente ajustables del modelo.

Una vez entrenado el conjunto de DT, una predicción en RF estará dada por realizar la predicción en cada DT y luego, en el caso de una regresión, promediar los resultados a fin de obtener la salida final. Para los problemas de clasificación, se utiliza una estrategia de votación, donde se promedian las probabilidades de pertenencia a cada clase predichas por cada DT y se predice la clase con la probabilidad más alta.

Al observar el algoritmo (ver Figura 2.11), es posible identificar su facilidad para ser paralelizado debido a la independencia de cada DT, lo que lleva a una reducción en los tiempos de entrenamiento. Sin embargo, a diferencia de los DT, los RF no son tan intuitivos, perdiéndose la interpretabilidad.

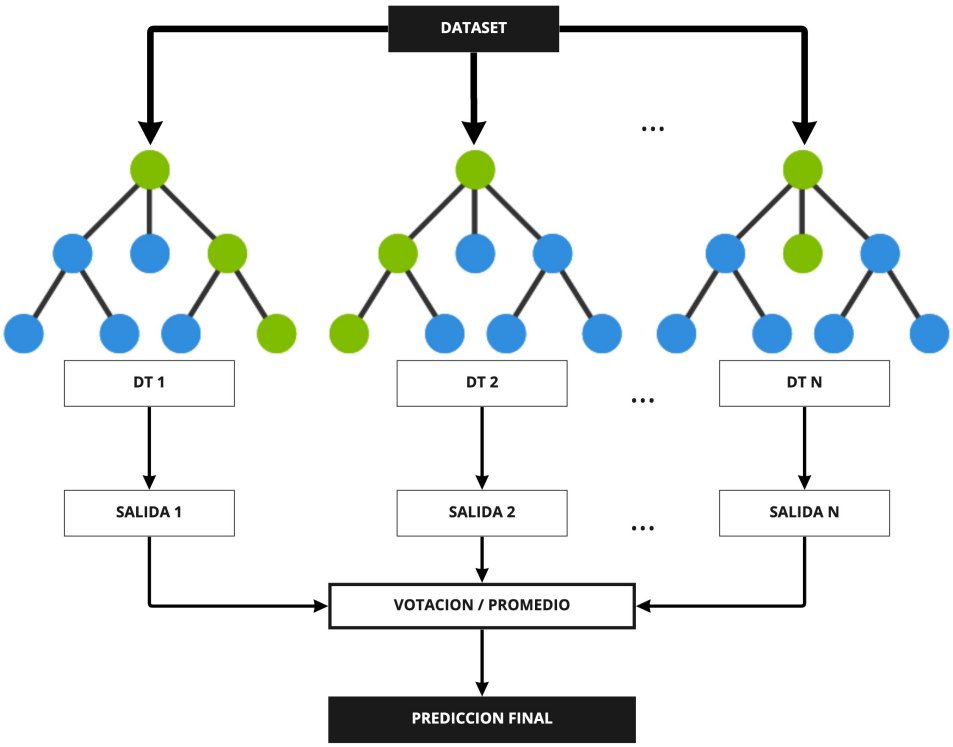


Figura 2.11: Proceso de predicción de un modelo RF.

2.2.4.5 K vecinos más cercanos

El algoritmo K vecinos más cercanos, también conocido como k-NN por sus siglas en inglés, se encuentra entre aquellos más simples [116]. Este método utiliza la cercanía del punto a predecir respecto a sus vecinos a fin de realizar una clasificación o regresión [66].

Su simpleza está dada por la etapa de entrenamiento donde únicamente se almacenan los ejemplos etiquetados (ver Pseudocódigo 1), $(x^{[i]}, y^{[i]}) \in D$ donde $(|D| = n)$, motivo por el cual también se lo denomina algoritmo de aprendizaje perezoso: el procesamiento y cálculo de los ejemplos de entrenamiento se pospone hasta que se realizan las predicciones o clasificaciones. Dado que depende en gran medida de la memoria para almacenar todos los datos de entrenamiento, también se lo denomina método de aprendizaje basado en instancias o basado en la memoria [116], [117]. Luego, a fin de realizar una predicción, se buscan los k vecinos más cercanos del punto a predecir y se calcula la etiqueta de clase (clasificación) o el valor continuo (regresión) respecto al valor de estos. Aunque la regla de predicción para k-NN en su forma más simple (k=1, el vecino más cercano o NN por sus siglas en inglés) es la misma tanto para la clasificación como para la regresión, en su forma general (k>1) existen dos algoritmos de predicción distintos:

- Para problemas de clasificación, se predice la clase destino como la etiqueta de clase que se presenta con mayor frecuencia dentro de los k vecinos cercanos: voto por pluralidad.
- Los problemas de regresión utilizan el mismo concepto general que los problemas de clasificación: primero, se buscan los k vecinos más cercanos en el conjunto de datos para luego realizar una predicción basada en las etiquetas de los k vecinos más cercanos. La única diferencia será que la predicción final es un valor real, con lo cual, un enfoque común es promediar los valores continuos de los k vecinos más cercanos.

De acuerdo al proceso mencionado, en lugar de diseñar un modelo global de aproximación, durante cada predicción, k-NN aproxima la función objetivo localmente dependiendo del punto de datos a predecir y su vecindad. A pesar de su simplicidad y desempeño, a medida que crece el conjunto de datos, la predicción de k-NN se vuelve cada vez más ineficiente, lo que compromete el rendimiento general del modelo [116], [117].

Pseudocódigo 1 Algoritmo de entrenamiento NN en el conjunto n-dimensional de entrenamiento $D(|D| = n)$

```
for each i = 1 to n do
    almacenar ejemplo de entrenamiento (x^[i], y^[i])
end for
```

Pseudocódigo 2 Algoritmo de predicción NN

```
closest_point=Null
closest_distance=∞
for each i = 1 to n do
    current_distance = d(x^[i], x^[q])
    if current_distance < closest_distance then
        closest_distance = current_distance
        closest_point = x^[i]
    end if
end for
```

La predicción para el punto $x^{[q]}$ en el Pseudocódigo 2 estará dada por el valor de la variable *closest_point*. Es posible identificar cómo el algoritmo realiza una búsqueda de vecinos por fuerza bruta, teniendo una complejidad de tiempo de O(N) por cada predicción a realizar. Es importante mencionar que existen diversas variantes y técnicas que permiten optimizar estos tiempos pudiéndose consultar en [116] ya que escapan del contenido de la presente investigación.

De esta forma, la implementación de k-NN se basa en dos aspectos fundamentales que pueden afectar a sus predicciones, rendimiento y desempeño:

- *Determinar la métrica de distancia*. Si bien existen muchas métricas de distancia a utilizar para seleccionar los k vecinos más cercanos, no existe una ideal y la elección depende en gran medida del problema. Para *features* continuos, la métrica de distancia probablemente más común es la distancia euclidiana, la cual computa la distancia en línea recta entre dos puntos $x^{[a]}$ y $x^{[b]}$: $d(x^{[a]}, x^{[b]}) = \sqrt{\sum_{j=1}^{m}(x_j^{[a]} - x_j^{[b]})^2}$. Otra opción popular es la distancia de Manhattan, $d(x^{[a]}, x^{[b]}) = \sum_{j=1}^{m}\left|x_j^{[a]} - x_j^{[b]}\right|$, que enfatiza las diferencias entre vectores de *features* distantes o valores atípicos a diferencia de la distancia euclidiana. Una desventaja de usar métricas de distancia más sofisticadas es que también suele tener un impacto negativo en la eficiencia computacional.

- *Definir k*. El valor k es un entero positivo que representa la cantidad de vecinos a verificar para determinar la clasificación o predicción de un punto dado. La mejor elección de k dependerá en gran medida de los datos de entrada: en general, un mayor valor reduce el efecto de los valores atípicos o ruidosos [118], pudiendo generar un sesgo alto y una varianza más baja, pero generando límites de clasificación menos claros [119]. Por otra parte, en la práctica, valores pequeños de k

suelen funcionar bien, pudiendo tener una varianza alta pero un sesgo bajo [66]. Definir k puede ser un acto de equilibrio ya que diferentes valores pueden llevar a un sobreajuste o un subajuste (2.2.5).

2.2.4.6 Redes Neuronales Artificiales

Hasta ahora se han visto modelos lineales, como los de las secciones (2.2.4.1), (2.2.4.2), los cuales son potentes y pueden ser aplicados a diversos problemas[17]. Sin embargo, como su nombre lo indica, estos modelos se encuentran limitados a representar únicamente relaciones lineales no pudiendo ser capaces de resolver problemas simples no lineales tal como el problema XOR [120]. A pesar de esto, estos modelos sirven como unidades básicas en la construcción de modelos no lineales más poderosos como son las Redes Neuronales Artificiales (ANN, por sus siglas en inglés). Las ANN son una familia de modelos computacionales, inspirados por las redes neuronales biológicas, quienes buscan simular el funcionamiento del cerebro humano [121]. La idea básica consiste en componer transformaciones (como son los modelos lineales) junto con funciones no lineales, denominadas funciones de activación, a fin de generar un mayor poder de predicción [122].

En general, las ANN pueden ser representadas a través de un grafo dirigido, donde cada nodo del mismo simboliza una neurona artificial y las aristas las conexiones entre ellas. Aunque pueden diferir en su arquitectura, el funcionamiento general de la red es el siguiente: cada neurona artificial es una unidad de cómputo que recibe n valores escalares como entrada, representados por $x_1, x_2, ..., x_n$, las cuales son ponderadas por una serie de valores llamados pesos, generalmente definidos como $w_1, w_2, ..., w_n$, correspondientes con los parámetros del modelo a optimizar (aristas del grafo). Luego, internamente se computa el estímulo total recibido dado la suma[18] de todas las entradas ponderadas. Finalmente, se aplica una función de activación[19] sobre el cálculo anterior generando el valor de salida de la neurona. La Figura 2.12 demuestra la idea general bajo el modelo ANN más simple, conocido como perceptrón [123], el cual es equivalente al modelo lineal LR (2.2.4.1).

[17] De hecho, generalmente sientan las métricas de desempeño base para la resolución de los mismos.

[18] Aunque la sumatoria es la operación comúnmente utilizada, podría ser reemplazada por alguna otra función de propagación tal como el máximo.

[19] Bajo la misma idea demostrada por el modelo LOR (2.2.4.2) al realizar una composición con la función sigmoidea.

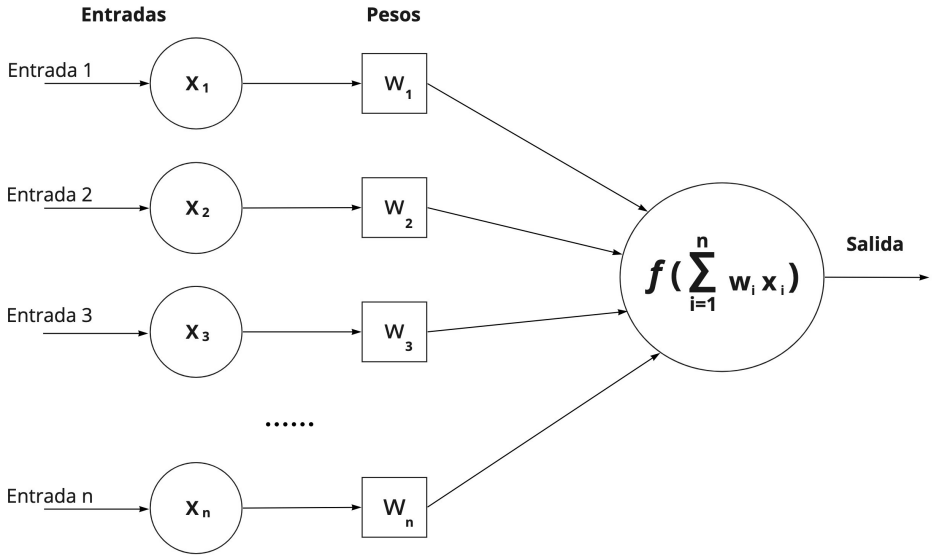


Figura 2.12: Esquema de un perceptrón con N neuronas de entrada.

Generalmente, las neuronas de una red se organizan en capas, las cuales reflejan el flujo de información. En la Figura 2.13 es posible identificar tres tipos de capas: una capa de entrada, que recibe los datos de entrada al modelo y los replica a la siguiente sin realizar procesamiento de estos; una o varias capas ocultas quienes reciben los estímulos de la anterior y los propagan a la siguiente; y una capa de salida, que representa la o las salidas de toda la red [124].

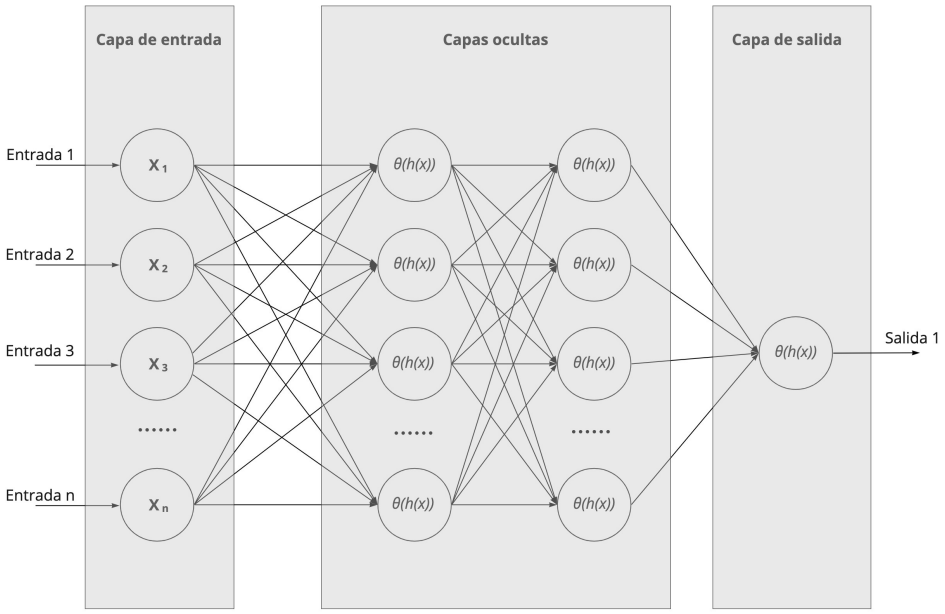


Figura 2.13: Esquema de un perceptrón multicapa o red *feedforward* con tres capas: la primera, de entrada, con N neuronas, luego dos capas ocultas, y finalmente una capa de salida, con una neurona.

La arquitectura de una ANN es un concepto más amplio que no solo abarca el número de capas sino el número de neuronas en cada una, la conexión entre ellas, su tipo e incluso la forma en que son entrenadas dando lugar a diferentes modelos ANN. Las capas ocultas abstraen patrones más generales permitiendo un aprendizaje más complejo, con lo cual la resolución de problemas más complejos amerita arquitecturas más complejas que el modelo perceptrón. El modelo anteriormente descrito en la Figura 2.13, utilizado en la presente investigación, se denomina perceptrón multicapa, donde:

- Cada neurona representa un perceptrón clásico independiente.
- Las salidas de las neuronas de cada capa se conectan como entradas en todas las neuronas de la siguiente capa (*fully-connected layers*).
- Existe al menos una capa oculta.
- Presenta una arquitectura *feedforward*, donde la información viaja en un sentido, desde la capa de entrada hacia la de salida a través de las capas ocultas sin ningún tipo de retroalimentación [62].

A fin de modelar distintos tipos de problemas, este modelo puede tener variaciones especificando la cantidad de capas y la cantidad de neuronas por cada una (hiperparámetros de la red). Los problemas de regresión o clasificación binaria pueden afrontarse utilizando una única neurona en la capa de salida. Asimismo, utilizando k neuronas de salida es

posible modelar un problema de clasificación multiclase. Esta versatilidad hace la diferencia respecto a los modelos lineales, siendo capaz de aproximar cualquier función continua sobre un subconjunto cerrado $\mathbb{R}^n$ [125], [126].

En lo que respecta a las funciones de activación, existe una gran variedad de funciones no lineales típicamente utilizadas [127], y su elección dependerá en gran medida de pruebas empíricas sobre el tipo de problema a resolver [128]. Además de la función sigmoidea ya vista, otras funciones popularmente utilizadas son:

- *Unidad Lineal Rectificada (ReLu, por sus siglas en inglés)*. Función muy simple donde para entradas negativas da una salida de 0, y para aquellas positivas se comporta como la función identidad. Se define como: $ReLU(x) = max(0, x)$.

- *Tangente Hiperbólica (Tanh)*. Función encargada de transformar su entrada en un valor dentro del intervalo abierto $(-1, 1)$, se define como: $Tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

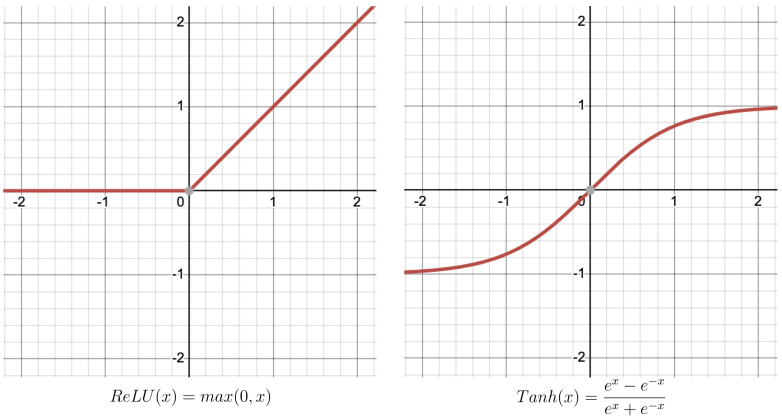


Figura 2.14: Gráfica de las funciones de activación ReLu y Tanh.

Las funciones de activación no poseen parámetros entrenables, únicamente son encargadas de romper la linealidad dando poder a la salida. En ocasiones, dependiendo el problema a modelar, las neuronas de la capa de salida utilizan una función de activación distinta del resto de las neuronas.

En general, en función del problema a resolver y el conjunto de datos de entrenamiento disponible, los algoritmos de entrenamiento de las ANN son responsables únicamente de minimizar el error respecto a los parámetros W (adaptar los pesos de la red). El algoritmo de aprendizaje comúnmente utilizado para entrenar las redes multicapas se denomina retropropagación o *backpropagation* [62], [129]. Este algoritmo supervisado realiza un proceso iterativo en dos fases basado en el algoritmo de descenso de gradiente visto anteriormente:

- *Propagación hacia adelante (forward pass)*. Las entradas de la red se utilizan como estímulo propagándose desde la capa de entrada, pasando por todas las capas ocultas, hasta generar la salida dada por la última capa. En este proceso, además

de realizar los cálculos, se almacenan tanto las variables intermedias como las salidas de la red para ser utilizadas en la siguiente fase. Finalmente, se calcula el error[20] para cada una de las neuronas de salida.

- *Propagación hacia atrás (backward pass)*. Durante esta fase, se atraviesa la red en orden inverso, desde la capa de salida a la de entrada, utilizando la regla de la cadena del cálculo [130] para computar y evaluar sucesivamente los gradientes, respecto a los parámetros de la red, a fin de minimizar el error global obtenido. Esto permite una ejecución computacionalmente poco costosa [62].

Diferentes tipos de problemas a resolver, cada uno con su complejidad, pueden requerir diferentes modelos de ANN con sus arquitecturas particulares. A pesar de estar por fuera del alcance de la presente investigación, existen otros modelos como lo son las redes convolucionales o las redes recurrentes [131], [132].

Finalmente, vale la pena mencionar que las definiciones algorítmicas expuestas en la presente sección (2.2.4) han sido introductorias y desde un marco teórico. Existen diferencias en la implementación de los distintos modelos de ML centradas en la optimización así como también el proceso de aprendizaje. En este último, una de las diferencias marcadas se encuentra en el algoritmo de descenso de gradiente: el crecimiento exponencial del conjunto de datos conlleva a un bajo desempeño del algoritmo dado por el costo en los cálculos de los gradientes como de la función de error [133]. Por esta razón, el algoritmo de aprendizaje utilizado en la práctica resulta ser una variante del descenso de gradiente llamada descenso de gradiente estocástico [134].

2.2.5 Evaluación de modelos

Durante el entrenamiento del modelo, el objetivo se centra en encontrar una función óptima que aproxime la distribución de los datos. De esta forma, el éxito del aprendizaje supervisado se mide al comparar las predicciones con las etiquetas verdaderas, pudiéndose observar el error cometido sobre el conjunto de datos de entrenamiento provisto [80]. Este proceso de ajuste del modelo para obtener el mejor rendimiento posible sobre el conjunto de entrenamiento es conocido como optimización [61]. Luego del entrenamiento, se avanza hacia la última etapa del *pipeline* general de ML presentado, la cual hace referencia a la evaluación.

Es importante y necesaria una evaluación del funcionamiento de los modelos entrenados, la cual permita determinar la necesidad de volver a iterar en el proceso hasta alcanzar el desempeño deseado. La evaluación radica en responder a la pregunta de: ¿Qué tan bueno es un modelo? Y particularmente ¿Qué tan bueno es con nuevos datos del mundo real o fuera del conocimiento del mismo? Esto desencadena en el deseo de modelos con capacidad de generalización a datos no vistos en el conjunto de entrenamiento [61].

El problema fundamental en ML consiste en generar una relación equilibrada entre la optimización y la generalización de los modelos. El objetivo será obtener una buena generalización optimizando el modelo a partir de los datos de entrenamiento [61].

[20]Típicamente se utilizan las mismas funciones de error utilizadas para los modelos lineales descritos.

Partiendo de esta idea, será necesario contar con al menos dos conjuntos de datos distintos de acuerdo a su contenido, aunque, respetando el mismo formato: el conjunto de entrenamiento y aquel de evaluación o prueba. El conjunto de evaluación introducido en el modelo entrenado, es utilizado para determinar su capacidad de generalización, o, lo que es lo mismo, decidir que tan bien ha aprendido comparando la salida predicha con la verdadera sin realizar modificación alguna sobre los parámetros [80].

Generalmente, en la práctica, se suele contar con un único conjunto de datos para el problema a resolver, el cual será dividido para el propósito mencionado. Existen diferentes formas de generar dicha división:

- *División fija*. El conjunto de datos se divide en un punto particular. Esto trae aparejado dos problemas: en primer lugar, quizás alguno de los conjuntos no sea del todo variado en cuanto a los ejemplos contenidos. En segundo lugar, el conjunto de datos podría estar ordenado y, por lo tanto, sesgar el entrenamiento y evaluación.

- *División estratificada*. Se toman los ejemplos de forma estratificada generando conjuntos de datos con cierta variabilidad aunque permanezca el problema de ordenamiento.

- *División estratificada aleatoria*. A la división anterior se agrega un reordenamiento aleatorio.

Aunque esta división suple la falta del conjunto de evaluación, introduce un problema dado por la reducción de la cantidad de datos disponibles para el entrenamiento, pero eso es algo con lo que se tendrá que convivir [80].

En la apertura de estos dos conjuntos de datos, se vislumbran dos conceptos importantes en lo que respecta a la evaluación de los modelos. Al comienzo del entrenamiento, en general, la optimización y la generalización se encuentran correlacionadas: a medida que disminuye el error en el conjunto de entrenamiento también se reduce el de evaluación. Mientras el error durante el entrenamiento sea alto, el modelo no es capaz de generalizar los patrones relevantes sobre los datos encontrándose en una condición de subajuste (*underfitting*). Por el contrario, a medida que crece el número de iteraciones en el entrenamiento es común que la capacidad de generalización del modelo se estanque o incluso se degrade mientras que el aprendizaje continúa mejorando: el modelo comienza a sobreajustarse (*overfitting*). Esto significa que aprende patrones específicos de los datos de entrenamiento que son irrelevantes cuando se trata de datos nuevos [61]. Este último concepto es análogo al estudio para un examen y la diferencia entre hacer un esfuerzo por comprender los conceptos generales o memorizar cada ejemplo del conjunto de tareas. El sobreajuste se asocia con aquel modelo que memoriza por completo los datos de entrenamiento proporcionados, generando un polinomio ajustado exactamente a todos los puntos (regresión) o con un límite de decisión excesivamente definido (clasificación). En fin, el objetivo para la resolución de problemas en el aprendizaje supervisado será crear modelos equilibrados entre la optimización y la generalización.

Es posible también considerar este problema como un balance entre la complejidad del problema y la del modelo entrenado. Para generalizar correctamente, a mayor com-

plejidad del problema, será necesario un modelo más complejo. En cambio, si el problema es relativamente sencillo y el modelo es sumamente complejo se caerá en un sobreajuste. Por el contrario, si el problema es complejo y el modelo es simple se tendrá un subajuste.

Cuando el conjunto de datos es pequeño, el proceso de división no sólo reduce drásticamente el conjunto de entrenamiento sino que puede conducir a un conjunto de evaluación que no abarque una muestra significativa de la distribución de los datos. Asimismo, realizar una única evaluación del modelo, usualmente, no es suficiente para concluir su capacidad de generalización u optimización considerando su entrenamiento ya que la misma dependerá de la división aleatoria particular de ambos conjuntos. Una solución habitual a estos problemas conduce a la utilización de un proceso llamado Validación Cruzada (CV, por sus siglas en inglés). En su enfoque básico, llamado *k-fold CV*, el conjunto de entrenamiento se divide en k conjuntos más pequeños llamados *folds*, y, luego, para cada uno de los k *folds* se itera realizando el siguiente procedimiento:

- Se entrena el modelo utilizando *k-1 folds* como conjunto de entrenamiento.
- El modelo resultante se evalúa con la parte restante de los datos (*fold k*).

Finalmente, la medida de generalización del modelo utilizando *k-fold CV* estará dada por el promedio de las evaluaciones realizadas en cada ciclo [61], [66]. Aunque existen otras versiones, como aquella estratificada o estratificada aleatoria, todas se basan en este procedimiento [66].

Existen diversas aproximaciones que permiten solucionar el problema del sobreajuste siendo la más simple la obtención de más datos: un modelo entrenado con una mayor cantidad de datos será capaz de generalizar mejor [61]. Sin embargo, como esto no siempre es posible, existe otro método que permite evitar el sobreajuste, restringiendo la complejidad del modelo, conocido como regularización. La técnica de regularización implica modificar el algoritmo de aprendizaje tal que el error de generalización se reduzca aunque no necesariamente lo haga el de entrenamiento [62]. Existen muchas estrategias de regularización, y, así como no existe un modelo ideal tampoco una mejor forma de regularizar [62]. Las principales técnicas de regularización utilizadas, conocidas como L1 y L2, agregan una penalización o costo a la función de error utilizada para el entrenamiento del modelo. Dada una función de costo L cuyo modelo depende de los parámetros W, la norma L1 modifica la función de error como: $L(W) = \frac{1}{n}\sum_i^n E_i + \lambda \sum_j^m \left|W_j\right|$ mientras que L2: $L(W) = \frac{1}{n}\sum_i^n E_i + \lambda \sum_j^m W_j^2$.

Cuanto mayor sea el valor de los pesos, más grande será la penalización y por consiguiente más grande el error. Como consecuencia, se intentará durante el aprendizaje forzar los parámetros W a tomar valores pequeños, lo que resulta en una distribución más regular de estos traduciéndose también en un modelo más simple. Asimismo, ambas formas de regularización se acompañan de un coeficiente λ que toma valores entre 0 y 1, el cual regula el efecto de las penalizaciones. Este último también es un hiperparámetro a encontrar.

En lo que respecta a las ANN, una de las formas más conocidas y utilizadas de regularización es la técnica de *dropout*, la cual se aplica a las capas de la red [135]. Esta

técnica se aplica durante el entrenamiento y consiste en ignorar aleatoriamente ciertas neuronas de la capa fijando sus valores de salida en cero. La proporción de neuronas a apagar es un hiperparámetro que se debe definir. Esta técnica evita que las predicciones de la red dependan de los pesos de neuronas específicas. Además, el ruido introducido en la red debido a la aleatoriedad de las neuronas ignoradas se traduce en la omisión de patrones casuales que la red memorizaría en caso de no estar este ruido presente [61], [128].

Independientemente del tamaño del conjunto de datos utilizado para probar el algoritmo entrenado, aún es preciso determinar si el modelo resultante es bueno o no. Aunque las funciones de error permiten cuantificar el error para luego optimizar el modelo, no son necesariamente útiles para expresar cuán bien está haciendo su tarea en términos de interpretabilidad. Para eso, a continuación se describe un conjunto de métricas destinadas a la evaluación y comparación de modelos, poniendo especial énfasis en aquellas aplicables tanto a problemas de clasificación binaria como de regresión por ser los abordados en la presente investigación.

2.2.5.1 Métricas de clasificación binaria

Una de las formas más comprensibles de representar el funcionamiento asociado a los modelos de clasificación radica en la utilización de una matriz de confusión [66]. Asociada a modelos de clasificación binaria, una matriz de confusión M es una matriz cuadrada de orden dos M_2, donde las filas representan la etiqueta verdadera de cada clase, mientras que las columnas representan la predicción del modelo entrenado. La diagonal principal simboliza el correcto funcionamiento del modelo, y lo que está por fuera de la misma son errores. La representación de la matriz de confusión para problemas de clasificación multiclase es la misma, modificando su orden en función de la cantidad de clases, pudiendo aplicarse las métricas desarrolladas a continuación bajo la misma idea [80].

La matriz de confusión de clasificación binaria se construye comparando para cada ejemplo del conjunto de datos, la etiqueta de clase predicha respecto a la actual de la muestra. La suma total de cada comparación se representa en una matriz M_2 dividida en 4 categorías representadas por sus siglas en inglés:

- *Verdaderos Positivos (TP)*. Resultados que el modelo predijo eran positivos y en efecto lo eran.
- *Verdaderos Negativos (TN)*. Resultados que el modelo predijo eran negativos y en efecto lo eran.
- *Falsos Negativos (FN)*. Resultados que el modelo predijo eran negativos, pero no lo eran.
- *Falsos Positivos (FP)*. Resultados que el modelo predijo eran positivos, pero no lo eran.

La Figura 2.15 muestra una representación gráfica de una matriz de confusión para un problema de clasificación binaria.

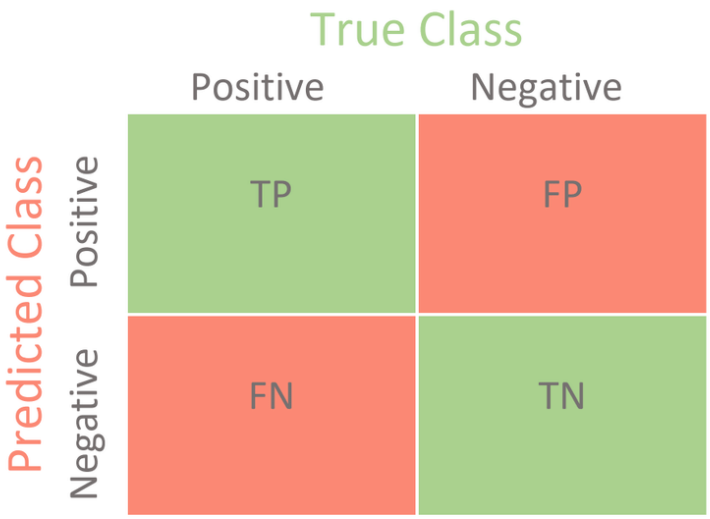


Figura 2.15: Matriz de confusión para un problema de clasificación binaria. Extraído de [136].

Aunque la matriz en sí misma no es una métrica, de ella se desprende el cálculo de ciertas métricas que resultan útiles para la evaluación de los modelos (ver Figura 2.16):

- *Accuracy*. Métrica más popular dentro de los modelos de clasificación, también conocida como tasa de acierto, se define como el porcentaje general de aciertos en clasificación del modelo, esto es, la predicción y el valor verdadero coinciden: $Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)}$.
- *Precision*. Se define como la cantidad de ejemplos reconocidos por el modelo como positivos que son realmente relevantes, es decir, eran efectivamente positivos: $Precision = \frac{TP}{(TP+FP)}$.
- *Recall*. Se define por aquellos ejemplos originalmente relevantes que fueron realmente identificados por el modelo como tales: $Recall = \frac{TP}{(TP+FN)}$.

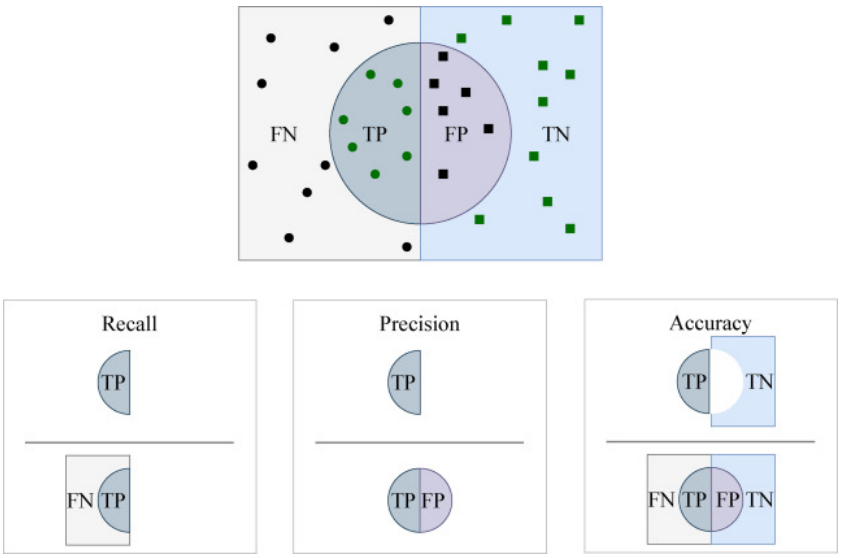


Figura 2.16: Visualización de *accuracy*, *precision* y *recall* como métricas de clasificación frecuentemente utilizadas. Extraído de [137].

Como es posible identificar, las métricas *precision* y *recall* tienen en cuenta el comportamiento del modelo respecto a la clase TP particular. Será posible obtener un *precision* perfecto cuando el modelo no haya tenido FP, y un *recall* perfecto cuando no se tengan FN. Sin embargo, la mayoría de las veces se suele querer visualizar ambas métricas en conjunto, aunque ciertas veces en determinados dominios se suele intentar elevar alguna de las métricas. De esta forma, a fin de generar una optimización balanceada entre estas métricas, se suele utilizar la métrica F_1score:

- F_1score. Se define como la media armónica entre las métricas *precision* y *recall*. Esto significa que cuando ambas métricas suben, F_1score sube; por el contrario, si alguna de las dos métricas baja, se penaliza el promedio entre las dos y baja el F_1score. Cuanto más cercano a 1 sea su valor entonces mejor será el desempeño del modelo; caso contrario, un valor cercano a 0 refleja el peor *precision* y *recall* posible: $F_1score = 2 \cdot \frac{(precision \cdot recall)}{(precision + recall)}$.

Durante la evaluación de los modelos es posible encontrar diferentes situaciones respecto al *accuracy* sobre el conjunto de datos:

- *Accuracy alto en entrenamiento y alto en evaluación*. Se tiene un modelo que realiza predicciones correctas para nuevos datos, el modelo aprendió a generalizar correctamente.
- *Accuracy alto en entrenamiento y bajo en evaluación*. El modelo se entrenó correctamente pero no realiza predicciones correctas para nuevos datos, lo cual da indicios de sobreajuste.
- *Accuracy bajo en entrenamiento*. El modelo no fue entrenado correctamente en función del conjunto de entrenamiento, lo cual da indicios de subajuste.

Una curva de entrenamiento representa una visualización del comportamiento del modelo (error) en cada iteración del aprendizaje. Lo deseable es que a medida que transcurran las iteraciones, el error disminuya, aunque no siempre la última iteración será el mejor modelo obtenido. El comportamiento del modelo en entrenamiento y evaluación produce dos curvas respectivas bajo sus nombres, las cuales pueden visualizarse en la Figura 2.17. Mientras que la curva de entrenamiento permite visualizar cómo el rendimiento del modelo cambia a medida que se aumenta la cantidad de datos de entrenamiento, la curva de evaluación muestra cómo el rendimiento del modelo cambia a medida que varía la complejidad del modelo. En general, a medida que aumenta la cantidad de datos de entrenamiento, el rendimiento del modelo mejora. Sin embargo, es importante tener en cuenta que puede haber un límite en cuanto a la cantidad de datos de entrenamiento que pueden mejorar significativamente el rendimiento del modelo. Por otra parte, a medida que aumenta la complejidad del modelo, el rendimiento mejora en el conjunto de datos de entrenamiento, aunque puede empeorar en el conjunto de datos de prueba debido al sobreajuste. La curva de evaluación puede ayudar a identificar el punto óptimo de complejidad del modelo que proporciona un buen rendimiento tanto en el conjunto de datos de entrenamiento como en el conjunto de datos de prueba.

La diferencia entre ambas curvas permite visualizar el comportamiento del modelo con nuevos datos, siendo generalmente un poco más alta la curva de evaluación en el error debido al desconocimiento de los datos. Asimismo, estas curvas también permiten visualizar errores en la configuración de los hiperparámetros de los modelos.

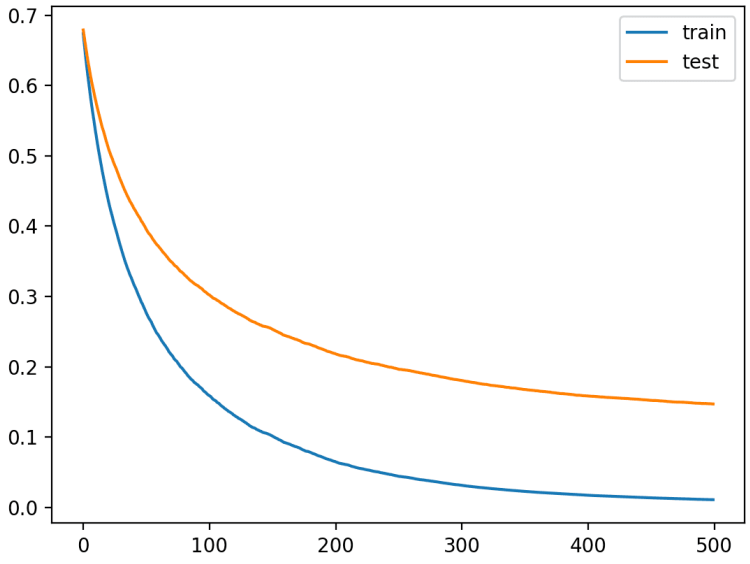


Figura 2.17: Gráfica de las curvas de entrenamiento (*train*) y evaluación (*test*).

Desbalanceo de clases Cuando un conjunto de datos posee proporciones de clase sesgadas entonces se encuentra desbalanceado: las clases que constituyen una mayor proporción se denominan clases mayoritarias, mientras que aquellas con una menor proporción se las conoce como clases minoritarias. Esto ocasiona que el modelo aprenda a predecir siempre lo mismo ya que durante el entrenamiento se busca minimizar cierto error medio respecto a todos los ejemplos. El resultado de esta condición se traduce en un *accuracy* elevado en contraposición al resto de las métricas y, es por lo que, la métrica de referencia en problemas de desequilibrio de clase es la $F_1 score$. Una primera solución para afrontar el desbalance de clases estará dada por una implementación específica de la función de error de forma tal que durante el entrenamiento del modelo se tenga en cuenta otra métrica. Sin embargo, en general, la mayoría de los modelos proporcionan un hiperparámetro que permite ponderar el error y otorgar más o menos peso a cada clase durante el entrenamiento para controlar el posible desbalance.

Curva ROC Dado que es posible utilizar las métricas mencionadas para evaluar un modelo de clasificación en particular, también es deseable poder comparar el mismo modelo bajo diferentes parámetros de aprendizaje o bien modelos completamente diferentes a fin de seleccionar aquel óptimo. En este caso, la curva ROC (*Receiver Operating Characteristic*) es útil [80], ya que grafica el comportamiento de un modelo binario respecto a la variación de su umbral de detección. Particularmente, la curva ROC es una técnica estándar que permite evaluar clasificadores binarios resumiendo la información de las matrices de confusión, generadas por cada variación del umbral, en un plano que describe el comportamiento de dos parámetros [80], [138]:

- El eje y muestra el *True Positive Rate* (TPR), definido de la misma forma que la métrica *recall*: $TruePositiveRate = Recall = \frac{TP}{(TP+FN)}$
- El eje x muestra el *False Positive Rate* (FPR), definido como: $FalsePositiveRate = 1 - especificidad = 1 - \frac{TN}{(FP+TN)}$.

De esta forma, la curva ROC se crea conectando los TPR y FPR para distintos valores del umbral. Una sola ejecución del modelo para un umbral determinado produce un único punto en la gráfica, siendo un clasificador perfecto el punto $(0,1)$, mientras que su polo opuesto estaría en $(1,0)$ traduciéndose a un modelo que se equivoca en todas sus predicciones. Esto vislumbra que cuanto más cerca de la esquina superior izquierda sea el comportamiento del modelo, mejor será su desempeño. La diagonal de $(0,0)$ a $(1,1)$ se comporta exactamente en el nivel de probabilidad desperdiciando esfuerzo de aprendizaje ya que una moneda lo haría igual de bien. La visualización ideal está dada por una curva alta suave o casi simulando un cuadrado en determinados conjuntos de datos, por el contrario, un comportamiento aleatorio del modelo se traduce en curvas raras. La curva ROC permitirá identificar aquel umbral de decisión más óptimo para un modelo dado.

A fin de realizar una comparación entre modelos, es posible calcular el Área Bajo la Curva (AUC, por sus siglas en inglés). La AUC resume a un valor único la curva ROC,

permitiendo fácilmente realizar comparaciones entre curvas, siendo aquella con AUC más cercana a 1 la mejor [138]. Asimismo, la clave para obtener una curva en lugar de un punto en el plano es utilizar CV [80]. En la Figura 2.18 puede observarse la gráfica de la curva ROC con los puntos descriptos anteriormente.

Curva Precision-Recall Aunque las curvas ROC pueden proporcionar un método robusto para identificar clasificadores potencialmente óptimos [138], también pueden ser optimistas bajo un desbalance de clase severo y pocos ejemplos de la clase minoritaria [139]. Debido a esto, una alternativa usual está dada por la curva *Precision-Recall* (PR) y su área bajo la curva (AUC). Estas curvas se computan de igual modo que las curvas ROC graficando el comportamiento del modelo para estas métricas. Como es posible observar en la Figura 2.18, la gráfica permite visualizar la relación de dependencia entre el *precision*, dado por el eje y, y *recall*, eje x, a medida que se modifican los umbrales de detección: a medida que aumenta el *recall* disminuye el *precision* y viceversa. Un modelo perfecto se representa como un punto $(1, 1)$ mientras que un modelo hábil está dado por una curva con inclinación hacia dicha coordenada. Un clasificador sin aprendizaje estará dado por una línea horizontal en el plano, con un *precision* proporcional a la cantidad de ejemplos positivos del conjunto de datos.

El enfoque de la curva PR en la clase minoritaria lo convierte en un diagnóstico eficaz para los modelos de clasificación binaria desbalanceados [138], [140]. De forma similar, las curvas PR pueden compararse sobre la base de su área.

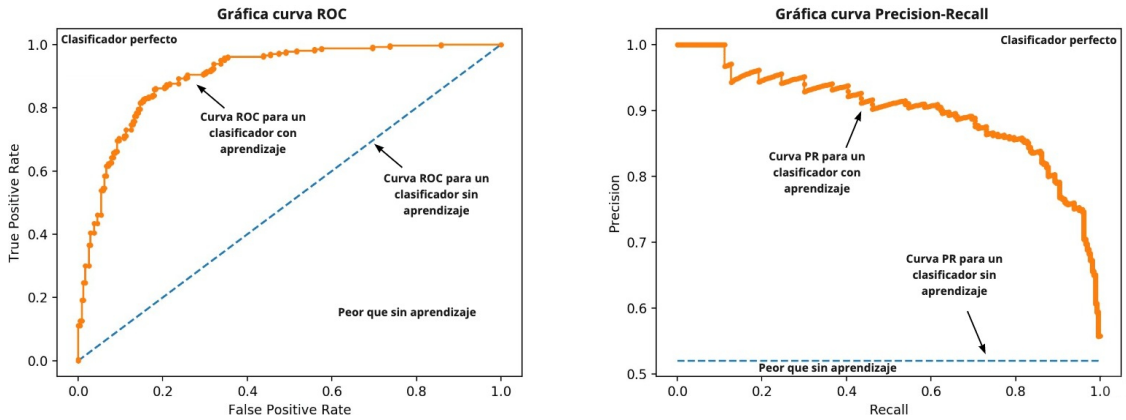


Figura 2.18: Gráfica de las curvas ROC y PR.

2.2.5.2 Métricas de regresión

Las métricas de evaluación para los modelos de regresión son diferentes a las métricas anteriormente presentadas, ya que ahora el resultado de la predicción será un rango continuo en lugar de un número discreto de clases. En los problemas de clasificación, la preocupación estaba dada por determinar si la predicción era correcta o no, siendo el *accuracy* una de sus métricas más populares. Sin embargo, como el *accuracy* es

una métrica de clasificación, no es posible calcularlo para un modelo de regresión. El desempeño de un modelo de regresión está dado como un error en sus predicciones. Por ejemplo, si el objetivo es predecir un valor numérico, no es preciso saber si el modelo predijo el valor exacto sino más bien qué tan cerca se encuentra dicha predicción del valor esperado. El error aborda exactamente esto y resume en promedio qué tan cerca se encuentran las predicciones de sus valores esperados. Las métricas de error presentadas a continuación se encuentran específicamente diseñadas para evaluar predicciones realizadas por modelos de regresión:

- *MSE*. Representa una de las métricas más populares en problemas de regresión y, como hemos visto en la sección (2.2.4.1), es la función de error utilizada en algoritmos de optimización. MSE se calcula como la media de las diferencias al cuadrado entre los valores predichos y esperados en el conjunto de datos (2.2.4.1). El cuadrado de su definición permite magnificar errores grandes, penalizando a los modelos al elevar la puntuación de error promedio.

- *Raíz de Error Cuadrático Medio (RMSE, por sus siglas en inglés)*. Representa la raíz cuadrada del MSE, teniendo la misma escala que la variable objetivo original. Cuanto menor sea su valor, mejor será la capacidad del modelo para ajustarse a los datos observados y realizar predicciones precisas. Es una medida popular debido a su capacidad para penalizar los errores grandes y su facilidad de interpretación.

- *Error Absoluto Medio (MAE, por sus siglas en inglés)*. A diferencia de la métrica anterior, MAE no magnifica los errores sino que la medida aumenta linealmente con los incrementos del error. El cálculo del MAE está dado por el promedio de los errores absolutos, esto implica que aunque la diferencia entre el valor esperado y predicho sea negativa, se fuerza a ser siempre positiva: $MAE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$, donde y_i representa el valor esperado mientras que $\hat{y}_i$ el valor predicho y n el tamaño del conjunto de datos.

- *Coeficiente de Determinación*. Usualmente denotado como R^2, representa la proporción de variación sobre la variable dependiente que es predecible a partir de la o las variables independientes en el modelo. Esta métrica proporciona una indicación de la calidad del aprendizaje y, por lo tanto, una medida de la probabilidad de que el modelo generalice correctamente, a través de la proporción de varianza explicada. Como la varianza depende del conjunto de datos, R^2 puede no ser significativamente comparable entre diferentes conjuntos de datos, siendo el mejor de los casos un modelo con $R^2 = 1$. Un modelo constante de referencia que siempre prediga el valor esperado (promedio) y que no tenga en cuenta los *features* de entrada, tendrá $R^2 = 0$. También, R^2 puede ser negativo cuando la media de los datos proporciona un mejor ajuste a los datos que las predicciones del modelo entrenado. Dado $\hat{y}_i$ (valor predicho para el ejemplo i) e y_i (el correspondiente valor verdadero), el Coeficiente de Determinación con n ejemplos se define como: $R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$, donde $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ es la media de los datos. La expresión de porcentaje de esta

medida genera una mayor intuición informativa en la evaluación de los modelos respecto a los rangos arbitrarios de las métricas anteriores.

2.2.5.3 Búsqueda de hiperparámetros

Los hiperparámetros [21] son aquellos que guían o configuran el algoritmo de optimización y la estructura de cada modelo de ML. Es importante mencionar que no existe una configuración ideal, sino que dependerá del modelo y el conjunto de datos. Por ejemplo, en las ANN la cantidad de capas ocultas y neuronas por capas, las funciones de activación utilizadas, el optimizador a utilizar y sus parámetros de control (como el *learning rate* para descenso de gradiente), la utilización de regularización con un valor de costo, las capas de *dropout*, son algunos de los hiperparámetros anteriormente mencionados, aunque existen muchos más.

A pesar de que se ha desarrollado como algoritmo de aprendizaje el descenso de gradiente, vale la pena mencionar la existencia de otros optimizadores con sus propios hiperparámetros [62] quedando por fuera del presente trabajo abordar en ellos.

Asimismo, aunque se conozcan los efectos generales de los hiperparámetros en un modelo, resulta un desafío establecer aquella combinación más óptima para un conjunto de datos dado. En general, el mejor enfoque busca objetivamente diferentes valores para los hiperparámetros, dentro de un conjunto finito, eligiendo aquel subconjunto que genere un modelo con el mejor desempeño para el conjunto de datos determinado. Este proceso se denomina optimización de hiperparámetros [61], [141].

2.3 Estado del arte

Aunque es posible encontrar publicaciones que desarrollen modelos predictivos para la identificación de personas con alto riesgo de padecer DM y PDM, la mayoría representa una prueba de concepto más que una implementación real por utilizar un conjunto de datos “de juguete”. Ejemplos de estos se pueden encontrar en [142] donde se utilizan cuatro clasificadores: *Naive Bayes* (NB), *Support Vector Machine* (SVM), RF y DT para predecir DM sobre el conjunto de datos PIMA [143] y en [144] donde se realiza una evaluación y comparación del comportamiento de los modelos de RF, DT y ANN sobre dicho conjunto de datos respecto a otro.

Por otra parte, como se ha mencionado en la sección 2.1.3, debido a que los estados de DM y PDM se encuentran condicionados por factores genéticos, ambientales y de comportamiento, las investigaciones realizadas sobre conjuntos de datos “reales” se corresponden a poblaciones diferentes a la de Argentina. En [145] se desarrolló un enfoque alternativo basado en SVM para detectar personas con DM y PDM sobre un conjunto de datos tomados de la Encuesta Nacional de Examen de Salud y Nutrición 1999-2004 (NHANES, por sus siglas en inglés) en la población de EE.UU., en [146] se compararon los modelos de LOR, ANN y DT para predecir DM y PDM utilizando factores

[21] Es importante no confundir este concepto con los parámetros de los modelos, que son los coeficientes internos aprendidos durante el entrenamiento sobre un conjunto de datos.

de riesgo comunes en la población de China. Asimismo, en [147] se desarrollaron dos modelos basados en ANN y SVM para la detección de PDM utilizando datos tomados de NHANES en la población de Corea. El caso más cercano se observó en [148] donde se compararon modelos (LOR, ANN, NB, k-NN y RF) para detectar DM no diagnosticada utilizando datos del Estudio Nacional Longitudinal de Salud de Adultos en Brasil (ELSA-Brasil). La mayoría de los modelos produjeron resultados similares y demostraron la viabilidad para la detección de esta enfermedad.

CAPÍTULO 3

PROPUESTA

En el presente capítulo, se abordará una descripción y caracterización del conjunto de datos así como el preprocesamiento efectuado sobre el mismo (Sección 3.1). Luego, se detallan las segmentaciones planteadas sobre el conjunto de datos disponible analizando las ventajas y desventajas de cada una (Sección 3.2). Finalmente, se especifican aquellas consideraciones a tener en cuenta antes de afrontar las experimentaciones propuestas (Sección 3.3).

3.1 Conjunto de datos

Como se ha mencionado en el capítulo anterior, el conjunto de datos (o *dataset*) resulta ser la pieza fundamental de cualquier modelo de ML por el conocimiento que puede ser extraído a través del proceso de aprendizaje. Bajo la premisa de *"intervenciones tempranas sobre el estilo de vida pueden prevenir o retrasar el desarrollo de DT2 en personas con TGA o GAA"* [18], el Centro de Endocrinología Experimental y Aplicada (CENEXA, UNLP-CONICET) desarrolló el programa PPDBA, con el objetivo de evaluar la efectividad de adoptar un estilo de vida saludable sobre la manifestación clínica de DT2 en personas con riesgo de desarrollarla.

El cuestionario FINDRISK [19] es un instrumento sencillo, útil y válido basado en un conjunto de 8 preguntas de fácil respuesta, donde cada una aporta un determinado puntaje [149]. Finalmente, la suma de todos los puntajes individuales anteriores se compara con cierto umbral, permitiendo así la identificación de personas con alto riesgo de desarrollar DT2 a 10 años. Si bien este cuestionario ha sido validado para la población Finlandesa, fue igualmente empleado por el PPDBA ante la ausencia de herramientas específicas para la población Argentina. Sobre esta base, el programa PPDBA confirmó la presencia de PDM y DM bajo la PTOG realizada en cada caso que superara el umbral definido [18].

El resultado de este programa derivó en el *dataset* a ser utilizado por los diversos modelos predictivos desarrollados en la presente investigación bajo la premisa de identificar personas con DM y PDM en Argentina.

3.1.1 Caracterización

Debido a la importancia que amerita, es vital entender el *dataset* disponible, y, para ello se realizó un análisis descriptivo inicial del mismo considerando toda su información. El conjunto de datos dispone de 1316 registros de personas, donde cada uno se corresponde a una persona que, mediante una PTOG, fue identificada como diabética, prediabética o sin ninguna de ellas. Además de datos de laboratorio (hemoglobina glucosilada; resultado de la PTOG; glucemia basal y postprandial; colesterol total, HDL y LDL; triglicéridos y creatinina), se cuenta con variables clínicas asociadas a los factores de riesgo agrupados en el cuestionario FINDRISK tales como el sexo; la edad; el Índice de Masa Corporal (IMC); la presión arterial; los antecedentes familiares de diabetes; los hábitos alimenticios y de actividad física; entre otros. La Tabla 3.1 presenta el conjunto de *features* disponibles, su descripción y categorización médica realizada con fines semánticos.

Tabla 3.1: Descripción del *dataset* completo.

Feature	Significado	Categoría médica
id	Identificador del registro	-
sexo	Sexo de la persona	Clínica
edad	Edad de la persona	Clínica
rango_edad	Agrupación de la edad en rangos	Clínica
imc	Índice de Masa Corporal calculado como $\frac{peso\_en\_kg}{altura\_en\_mts^2}$	Clínica
rango_imc	Agrupación del IMC en rangos	Clínica
circunferencia _de_cintura	Circunferencia de cintura expresada en centímetros	Clínica
rango_circunferencia _de_cintura	Agrupación de la circunferencia de cintura en rangos	Clínica
actividad_fisica	Práctica regular de actividad física	Clínica
consume_vegetales _frutas_y_hortalizas	Incluye frutas y verduras en su alimentación diaria	Clínica
toma_medicacion _para_controlar_hta	Toma medicación para controlar hipertensión	Clínica
le_encontraron _hiperglucemia	Detección de hiperglucemia en un examen médico, durante embarazo o estudio	Clínica
le_diagnosticaron_diabetes _a_algun_familiar	Antecedentes familiares de DM	Clínica
glucemia_basal	Valor de glucemia en ayunas expresado en mg/dL	Laboratorio
glucemia_postprandial	Valor de glucemia, expresado en mg/dL, 2 horas después de haber ingerido una solución azucarada (75g de glucosa disuelta en 375ml de agua)	Laboratorio
colesterol_ldl	Colesterol LDL expresado en mg/dL	Laboratorio
colesterol_total	Colesterol total expresado en mg/dL	Laboratorio
colesterol_hdl	Colesterol HDL expresado en mg/dL	Laboratorio
trigliceridos	Triglicéridos expresados en mg/dL	Laboratorio
creatinina_basal	Creatinina basal expresada en mg/dL	Laboratorio
hemoglobina_glucosilada	Hemoglobina glucosilada expresada en porcentaje	Laboratorio
resultado_ptog	Resultado de la PTOG	Laboratorio
clase	Resultado agrupado de la PTOG	-

De los 1316 registros actuales, 80 debieron ser eliminados ya que omitían valores requeridos de glucemia para poder calcular el resultado de la PTOG. En consecuencia, no es posible determinar la `clase` a la que pertenece la persona.

El proceso exploratorio inicial realizado permitió extraer cierta información la cual

se centralizó en las siguientes tablas (Tabla 3.2 y Tabla 3.3) y gráficos (Figura 3.1 y Figura 3.2). Su separación, además de permitir una distinción de los *features* por su tipo, brinda información representativa a cada uno.

Tabla 3.2: Exploración inicial de los *features* categóricos asociados al *dataset* completo.

Feature	Tipo	N (no nulo)	Porcentaje de nulos (N)	Valor	N	Porcentaje
sexo	Cualitativo (Nominal)	1236	0%	0 ⇒ Femenino (f)	841	68%
				1 ⇒ Masculino (m)	395	32%
rango_edad	Cualitativo (Ordinal)	1236	0%	0 ⇒ menor de 45 años	205	17%
				1 ⇒ entre 45 y 54 años	435	35%
				2 ⇒ entre 55 y 64 años	429	35%
				3 ⇒ mayor de 64 años	167	14%
rango_imc	Cualitativo (Ordinal)	1236	0%	0 ⇒ menor de 25 IMC	93	8%
				1 ⇒ entre 25 y 30 IMC	319	26%
				2 ⇒ mayor de 30 IMC	824	67%
rango___cintura	Cualitativo (Ordinal)	1236	0%	0 ⇒ m: menos de 94cm	56	5%
				0 ⇒ f: menos de 80cm		
				1 ⇒ m: entre 94cm y 102cm	205	17%
				1 ⇒ f: entre 80cm y 88 cm		
				2 ⇒ m: mayor de 102 cm	975	79%
				2 ⇒ f: mayor de 88cm		
actividad_fisica	Cualitativo (Nominal)	1236	0%	0 ⇒ No	915	74%
				1 ⇒ Sí	321	26%
cons___hortalizas	Cualitativo (Nominal)	1236	0%	0 ⇒ No todos los días	821	66%
				1 ⇒ Todos los días	415	34%
toma___hta	Cualitativo (Nominal)	1236	0%	0 ⇒ No	497	40%
				1 ⇒ Sí	739	60%
le___hiperglucemia	Cualitativo (Nominal)	1236	0%	0 ⇒ No	999	81%
				1 ⇒ Sí	237	19%
le_diag___familiar	Cualitativo (Nominal)	1233	0,2% (3)	0 ⇒ No	395	32%
				1 ⇒ Sí: abuelo, tío, tía, o primo	412	33%
				2 ⇒ Sí: padre, hermano/a e hijo/a	426	35%
resultado_ptog	Cualitativo (Nominal)	1236	0%	0 ⇒ Normal	620	50%
				1 ⇒ GAA	300	24%
				2 ⇒ TGA	74	6%
				3 ⇒ GAA+TGA	106	9%
				4 ⇒ DM	136	11%
clase	-	1236	0%	0 ⇒ Normal	620	50%
				1 ⇒ PDM (GAA, TGA o GAA+TGA)	480	39%
				2 ⇒ DM	136	11%

Tabla 3.3: Exploración inicial de los *features* continuos asociados al *dataset* completo.

Feature	Tipo	N (no nulo)	Porcentaje de nulos (N)	Media	Desviación estándar	Mínimo	Máximo
edad	Cuantitativo (Discreto)	672	45,6% (564)	57,237	8,849	24	99
imc	Cuantitativo (Continuo)	617	50,1% (619)	31,651	6,337	17,2	54,9
cir___cintura	Cuantitativo (Discreto)	55	95,6% (1181)	101,309	13,684	62	127
glucemia_basal	Cuantitativo (Continuo)	1236	0%	104,362	27,298	45	482
glucemia_postprandial	Cuantitativo (Continuo)	1181	4,4% (55)	119,594	42,532	15	343
colesterol_ldl	Cuantitativo (Continuo)	521	57,8% (715)	119,798	36,851	0	265
colesterol_total	Cuantitativo (Continuo)	531	57% (705)	198,271	41,148	87	369
colesterol_hdl	Cuantitativo (Continuo)	530	57,1% (706)	49,826	14,419	15	189
trigliceridos	Cuantitativo (Continuo)	531	57% (705)	151,409	95,687	37	983
creatinina_basal	Cuantitativo (Continuo)	617	50,1% (619)	1,117	5,809	0,21	118
hem___glucosilada	Cuantitativo (Continuo)	601	51,4% (635)	5,614	0,440	4,1	7,4

La información exploratoria resumida en de las Tablas 3.2 y 3.3 y los gráficos presentados en las Figuras 3.1 y 3.2 permite identificar ciertas cuestiones:

- Existen varios *features* continuos que presentan valores nulos, mientras que `le_diagnosticaron_diabetes_a_algun_familiar` es el único de los *features* categóricos que presenta un porcentaje de nulidad. Los valores nulos y su tratamiento se analizan con mayor profundidad en la Sección 3.1.2.
- La mayoría de las personas se encuentra dentro del grupo etario de 45-64 años.
- La mayoría de las personas tienen un IMC mayor a 30 kg/m2.
- Hay más personas del sexo femenino que del masculino.
- La mayoría de las personas tienen una circunferencia de cintura de más de 102cm y de 88cm para el sexo masculino y femenino, respectivamente.
- La mayoría de las personas no realizan actividad física regular en su rutina diaria; no consumen vegetales, frutas y hortalizas; dependen de medicación para controlar la hipertensión; no han experimentado hiperglucemia en sus controles médicos previos; y tienen antecedentes familiares cercanos, ya sea de primer o segundo grado, que padecen DM.
- Los *features* asociados a la glucemia basal, colesterol HDL, triglicéridos y creatinina basal parecieran tener una amplia dispersión; el resto no.

- En cuanto a la `clase`, la mitad de las personas no están en riesgo de padecer PDM o DM. Y de la mitad restante, el porcentaje de personas con PDM es superior sobre aquellas con DM. En caso de no agrupar los pacientes con PDM o DM, es posible visualizar un desbalance de clases a tener en cuenta.

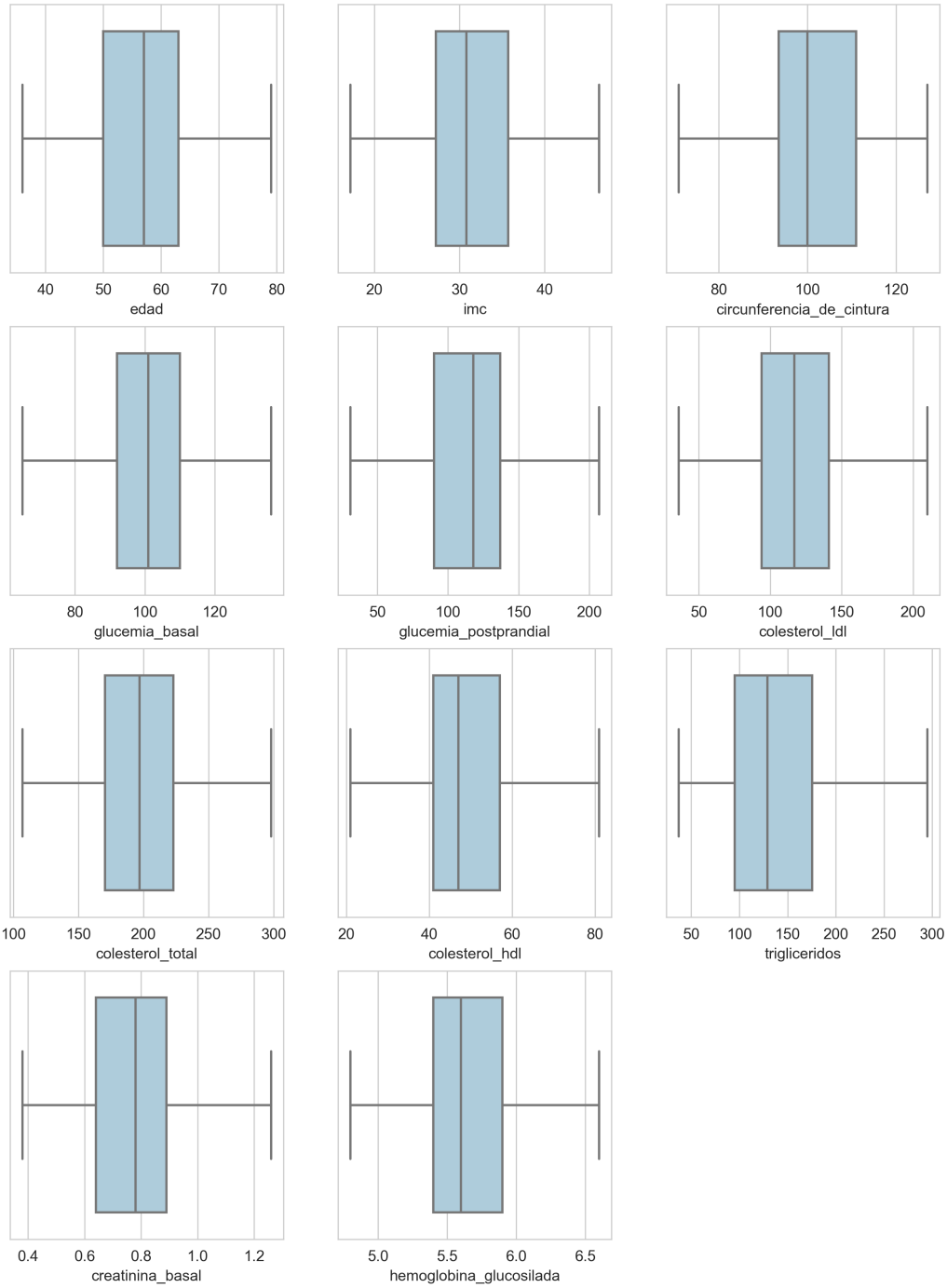


Figura 3.1: Diagramas de cajas de los *features* continuos asociados al *dataset* completo.

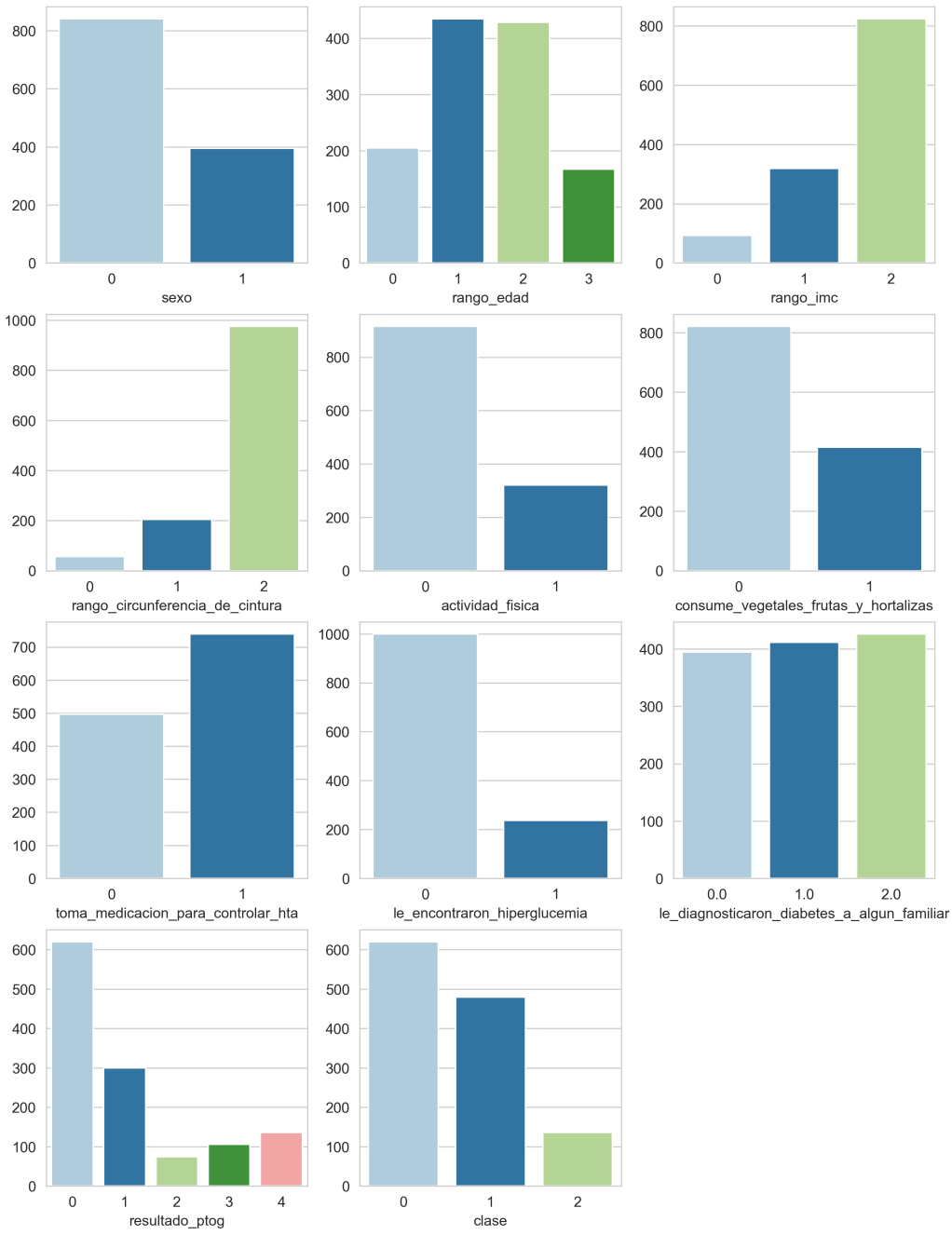


Figura 3.2: Histogramas de los *features* categóricos asociados al *dataset* completo.

3.1.2 Preprocesamiento

A continuación se describe el preprocesamiento realizado sobre el dataset original:

3.1.2.1 Eliminación de variables

Como es posible observar en la Tabla 3.1, el dataset inicial consta de 23 features. Sin embargo, en primera instancia, se encontraban ciertas variables de control (`finrisk_id`, `ptog_id` y `origen_datos`) las cuales fueron eliminadas por no aportar valor al propósito de la presente investigación. Asimismo, la variable `id` también fue excluida por los mismos motivos. Por otra parte, las variables `edad`, `imc` y `circunferencia_de_cintura` también fueron eliminadas por su alto porcentaje de valores nulos: en su lugar se decidió utilizar las variables categóricas asociadas que contemplan rangos de valores.

Las PTOG son procedimientos costosos desde el punto de vista económico, además de resultar engorrosos para los pacientes. Es por lo que disponer de un modelo que pueda identificar DM no diagnosticada o PDM mediante parámetros clínicos, como la edad o IMC, junto a variables de laboratorio de rutina, como la glucemia o el colesterol total, representaría una herramienta valiosa frente a los mecanismos actuales: se podrían disminuir drásticamente el número de las PTOG, incluso la realización de otros exámenes de laboratorios si el paciente dispusiera de datos recientes. Por tal motivo, la variable `glucemia_postprandial` no se tuvo en cuenta en ningún experimento, ya que implicaría que la persona tuviera que hacerse una PTOG, careciendo de sentido para el modelo planteado.

Por otra parte, la variable `resultado_ptog`, la cual representa el resultado de dicha prueba, fue eliminada ya que la discriminación de los diferentes tipos de PDM escapa de los límites de la presente investigación. Como es posible observar, el *feature* de `clase` realiza una agregación de GAA, TGA y GAA+TGA en PDM.

3.1.2.2 Valores atípicos

A fin de analizar la presencia de ruido de las variables restantes del proceso de eliminación, se utilizó el método de Tukey para identificar los intervalos de valores atípicos leves y extremos. La Tabla 3.4 muestra los valores mínimos, máximos, cuartiles, RIC, bigotes y límites para los *features* continuos del *dataset* a tener en cuenta. Asimismo, la Figura 3.3 presenta los diagramas de caja asociados con estos *features*.

A continuación, considerando lo calculado anteriormente, se procedió a examinar los valores de las variables para listar los valores atípicos, los cuales se detallan en la Tabla 3.5. En ella, se puede notar que todas presentan valores atípicos leves, mientras que sólo 4 presentan valores atípicos extremos (`glucemia_basal`, `colesterol_hdl`, `trigliceridos` y `creatinina_basal`).

Tabla 3.4: Análisis de valores atípicos para los *features* continuos a considerar.

Feature	Min	Max	Q1	Q2	Q3	RIC	Límite inferior	Límite superior	Bigote inferior	Bigote superior
glucemia_basal	45	482	92	101	110	18	65	137	65	136
colesterol_ldl	36	265	94	117	141	47	23,5	211,5	36	210
colesterol_total	87	369	170,5	197	223	52,5	91,75	301,75	107	298
colesterol_hdl	15	189	41	47	57	16	17	81	21	81
trigliceridos	37	983	95	129	175,5	80,5	-25,75	296,25	37	295
creatinina_basal	0,21	118	0,64	0,78	0,89	0,25	0,26	1,27	0,38	1,26
hem___glucosilada	4,1	7,4	5,4	5,6	5,9	0,5	4,65	6,65	4,8	6,6

Tabla 3.5: Valores atípicos leves y extremos para los *features* continuos a considerar. Los valores resaltados en **negrita** son los considerados como atípicos por expertos en el dominio médico.

Feature	Intervalos de valores típicos leves	Atípicos leves	Atípicos extremos
glucemia_basal	[38; 65; 137; 164]	36 registros	24 registros
colesterol_ldl	[-47; 23,5; 211,5; 282]	[239; 230; 242; 265; 217; 254; 231]	-
colesterol_total	[13; 91,75; 301,75; 380,5]	[323; 317; 304; 369; 335; 303; 333; 87; 311]	-
colesterol_hdl	[-7; 17; 81; 105]	[100; 91; 92; 82; 88; 85; 84; 15; 92]	[143; 189]
trigliceridos	[-146,5; -25,75; 296,25; 417]	21 registros	12 registros
creatinina_basal	[-0,11; 0,26; 1,27; 1,64]	[1,56; 1,37; 0,21; 1,29; 1,4; 1,32; 1,49; 1,3]	[2,57; 1,77; **118**; 1,73; **85**; 2,17]
hem___glucosilada	[3,9; 4,65; 6,65; 7,4]	23 registros	-

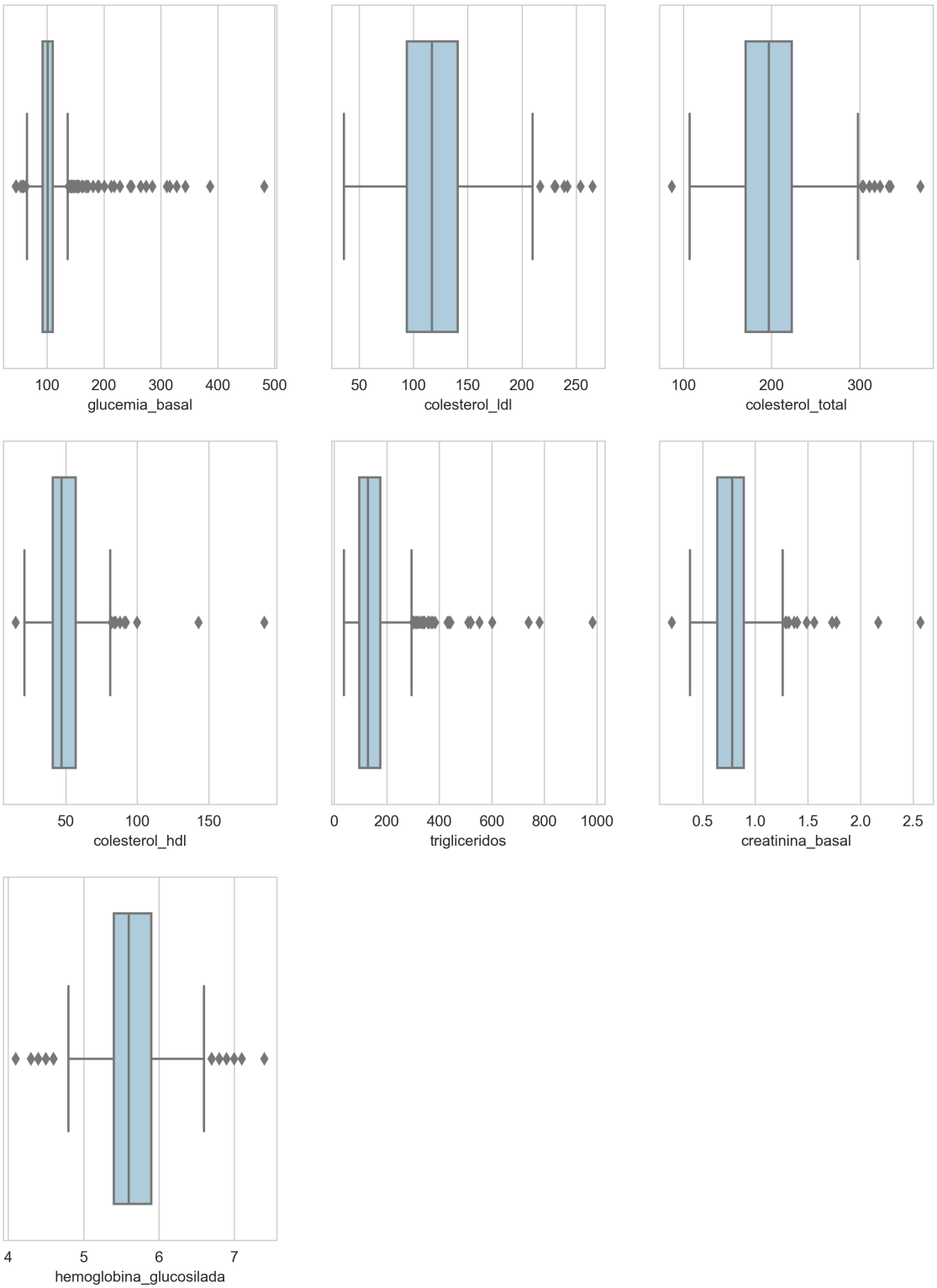


Figura 3.3: Diagramas de caja para la identificación de valores atípicos sobre aquellos *features* continuos asociados al *dataset* completo disponible.

Se procedió a consultar con expertos del dominio médico sobre las ocurrencias de valores atípicos en las determinaciones de laboratorio. Se concluyó que, si bien son valores que estadísticamente pueden ser considerados atípicos, sí se encuentran dentro de los posibles valores extremos para estas determinaciones. La excepción a lo anterior son los valores 85 y 118 en `creatinina_basal` que efectivamente corresponden a (posibles) errores de carga. Por lo tanto, se procedió a reemplazar con nulos dichos valores particulares. Como consecuencia, la Tabla 3.6 muestra los valores asociados a la `creatinina_basal` luego de su preprocesamiento.

Tabla 3.6: Creatinina basal luego de la eliminación de sus valores atípicos.

Feature	Tipo	N (no nulo)	Porcentaje de nulos (N)	Media	Desviación estándar	Mínimo	Máximo
creatinina_basal	Cuantitativo (Continuo)	615	50,2% (621)	1,117	5,809	0,21	118

3.1.2.3 Valores Nulos

Realizando un análisis en lo que respecta al porcentaje de nulidad observado de la Tabla 3.2, `le_diagnosticaron_diabetes_a_algun_familiar`, es el único que presenta valores incompletos (3 registros). No ocurre lo mismo con los *features* continuos, Tabla 3.3, donde son más aquellos que no presentan valores:

- El cuestionario FINDRISK refiere a los rangos de edad, IMC y circunferencia de cintura; sin embargo, algunas personas completaron con valores exactos en lugar de seleccionar el rango. Por tal motivo, los *features* de `edad`, `imc` y `circunferencia_de_cintura` poseen un gran porcentaje de valores nulos, siendo el último aquel con mayor porcentaje de nulidad entre todos los *features* disponibles. El resto, provenientes del cuestionario, no poseen valores nulos.

- Sobre todas las personas se realizó la PTOG, con lo cual, el *feature* `glucemia_basal` se encuentra completo. Sin embargo, los *features* de laboratorio restantes presentan valores nulos: `glucemia_postprandial`, `colesterol_total`, `colesterol_ldl`, `colesterol_hdl`, `trigliceridos`, `creatinina_basal` y `hemoglobina_glucosilada`.

Como se ha mencionado en la Sección 2.2.3.1, existen algunas alternativas para el tratamiento de variables con valores nulos, cada una con su costo-beneficio:

- Eliminación de registros completos, lo que lleva a disminuir el tamaño de la muestra. Para el *dataset* utilizado en la presente investigación, tomar este camino significaría reducir de 1236 a 506 registros, manteniendo los 16 *features*.

- Eliminación por columna, lo que lleva a tener menos *features* durante el entrenamiento de los modelos. Para el *dataset* utilizado en la presente investigación, tomar este camino significaría reducir de 16 a 9 *features*, manteniendo los

1236 registros. Si se presta atención, se estaría eliminando también el *feature* `le_diagnosticaron_diabetes_a_algun_familiar` que tan solo posee 3 ejemplos nulos.

- Reemplazar los valores nulos con algún valor especial que tenga sentido considerando el dominio, como la media, la mediana o la moda. Esta alternativa se descarta, ya que sería poco probable que el valor especial fuese representativo para la persona, especialmente para las variables de laboratorio.
- Utilizar alguna técnica basada en ML que permita calcular el valor probable. Esta alternativa podría intentarse pero su implementación queda por fuera del alcance de la presente investigación.

En primera instancia, se decidió utilizar la eliminación de registros completos únicamente para el *feature* `le_diagnosticaron_diabetes_a_algun_familiar`, que cuenta con sólo 3 ejemplos nulos. Para el resto, se decidió tomar diferentes alternativas con el objetivo de evaluar y obtener el mejor *dataset* dentro de la información disponible para los modelos evaluados. En la Sección 3.2 se detallarán cada una de las segmentaciones propuestas.

3.1.2.4 Binarización de la clase

De la Tabla 3.2 y la Figura 3.2 es posible notar que la distribución de clases no se encuentra equilibrada (*feature* `clase`). Para minimizar el impacto de esta cuestión, se procede a crear una nueva variable de clase que divida entre personas sin riesgo de tener PDM o DM (registros con valor "Normal") y las que sí lo tienen (registros con valor "PDM" o "DM"). Como resultado, el *feature* queda balanceado en cuanto a ocurrencias de cada valor y, a la vez, se simplifica el análisis posterior al pasar a ser ahora un problema de clasificación binaria (ver Tabla 3.7). Sin embargo, se deberá ser cuidadoso al momento de analizar los resultados, especialmente con lo que se pueda decir sobre predicción de DM, por ser la de menor ocurrencia.

Tabla 3.7: Variable de clase creada transformando el problema en uno de clasificación binaria.

Feature	Tipo	N (no nulo)	Porcentaje de nulos (N)	Valor	N	Porcentaje
clase	-	1236	0%	0 ⇒ Sin riesgo	620	50%
				1 ⇒ Con riesgo	616	50%

Asimismo, también se decidió crear otra nueva variable (`clase_desc`) a ser utilizada con fines descriptivos y representativos durante la evaluación de los modelos. La misma representa la descripción de la clase, que, para un problema de clasificación binaria: "Sin riesgo" representa valores con `clase` en 0 y "Con riesgo" aquellos con `clase` en 1. Por otra parte, considerando un problema de regresión: "Normal" representa valores con `clase` en 0, "PDM" con `clase` en 1 y "DM" aquellos con `clase` en 2.

3.1.3 Correlaciones

La Figura 3.4 muestra la matriz de correlación obtenida sobre el *dataset* inicial disponible. De la matriz, se derivan las siguientes conclusiones:

- Existe una correlación lineal débil[1] entre el rango de la circunferencia de cintura y el rango IMC, lo que tiene sentido desde el punto de vista clínico.
- También tiene sentido que existan correlaciones débiles entre los rangos de edad, IMC y circunferencia de cintura y sus valores asociados en los *features* de `edad`, `imc` y `circunferencia_de_cintura`.
- El `colesterol_total` se calcula a partir del `colesterol_ldl`, `colesterol_hdl` y los `trigliceridos` [150]. Particularmente, el `colesterol_ldl` y `colesterol_hdl` son colineales, lo que explica la correlación fuerte entre el `colesterol_total` y `colesterol_ldl` con lo cual es de suponer su correlación.
- Asimismo, la relación entre la `glucemia_basal` y la `glucemia_postprandial` tiene sentido ya que la segunda se mide unas horas después que la primera.
- La `glucemia_basal` y `hemoglobina_glucosilada` están cerca de tener una correlación débil. Su explicación sienta las bases en una fórmula plateada en [151] donde:

$$hemoglobina\_glucosilada = (glucemia\_basal + 77{,}3)/35{,}6 \tag{3.1}$$

$$glucemia\_basal = (35{,}6 * hemoglobina\_glucosilada) - 77{,}3 \tag{3.2}$$

- La `creatinina_basal` no tiene ningún tipo de correlación con el resto de los *features* de laboratorio.
- Entre la `glucemia_basal` y `resultado_ptog` se tiene una correlación débil, lo que tiene sentido considerando que la glucemia basal es uno de los valores que definen el resultado de la PTOG.
- En el resto de los casos, no hay correlaciones lineales, aunque la `clase` es una agrupación a partir del `resultado_ptog`, de ahí su fuerte correlación.

El `imc` y la `circunferencia_de_cintura` presentan un porcentaje de nulidad elevado, los cuales se conjugan con aquellos datos de laboratorio que presentan valores faltantes. Esto produce que, en la Figura 3.4, se observen zonas de tonalidad gris, las cuales se corresponden con aquellos casos donde no existe un valor de correlación. Esta ausencia es consecuencia de valores faltantes o indefinidos en el conjunto de datos, dado que los coeficientes de correlación se basan en la información disponible para su cálculo.

[1] Sean A y B, poseen una correlación lineal débil si $0{,}5 \leq |Corr(A,B)| \leq 0.8$ y una correlación lineal fuerte si $|Corr(A,B)| > 0.8$. Por el contrario, A y B no se encuentran correlacionadas si $|Corr(A,B)| < 5$.

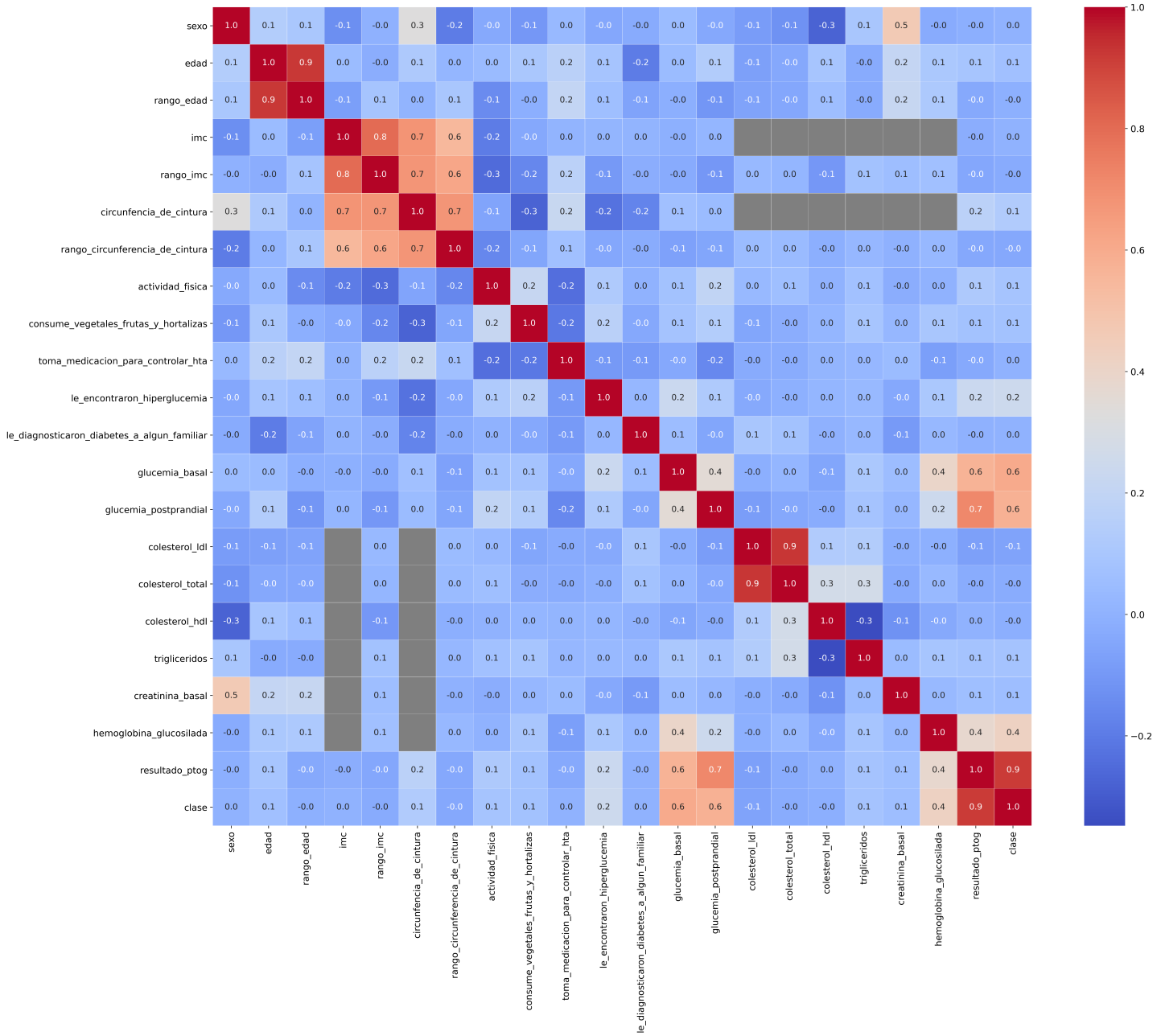


Figura 3.4: Matriz de correlación de todas las variables asociadas al *dataset* original. Las zonas de tonalidad gris son consecuencia de valores faltantes.

3.2 Segmentaciones propuestas

En resumen, el *dataset* original disponible cuenta con 1236 ejemplos y 25 *features*, sin contar la `clase`, los cuales pueden dividirse en:

- *Features clínicos (9)*. `sexo`, `rango_edad`, `rango_imc`, `rango_circunferencia_de_cintura`, `actividad_fisica`, `consume_vegetales_frutas_y_hortalizas`, `toma_medicacion_para_controlar_hta`, `le_encontraron_hiperglucemia`, `le_diagnosticaron_diabetes_a_algun_familiar`.
- *Features de laboratorio (7)*. `glucemia_basal`, `colesterol_ldl`, `colesterol_total`, `colesterol_hdl`, `trigliceridos`, `creatinina_basal`, `hemoglobina_glucosilada`.
- *Features eliminados (9)*. `id`, `edad`, `imc`, `circunferencia_de_cintura`, `glucemia_postprandial`, `resultado_ptog`, `finrisk_id`, `ptog_id`, `origen_datos`.

Como se detalló en la Sección 1.2, este trabajo final propone desarrollar y evaluar modelos predictivos que permitan identificar personas con DM y PDM considerando como base de datos la correspondiente al programa PPDBA. Luego del preprocesamiento mencionado en la Sección 3.1.2, se concluye en un *dataset* con 16 *features* y 1233 ejemplos. En lugar de plantear un único escenario donde se considere el conjunto de datos disponible, se decidió efectuar distintas segmentaciones o subdivisiones del mismo. El propósito de este particionamiento consiste en generar y evaluar modelos que presenten diferentes relaciones de costo-beneficio.

3.2.1 *Datasets* con información clínica y de laboratorio

En primer lugar, se considera toda la información disponible (datos clínicos asociados a factores de riesgo junto a datos de laboratorio). Debido a que los *features* de laboratorio poseen un porcentaje de nulidad alto, se planteó la posibilidad de generar varios *datasets* a partir de este, cada uno tomando distintos criterios para el tratamiento de registros nulos:

- `Dataset Clínica+Laboratorio (DCL)`. Conjunto de datos al cual se realizó una eliminación de registros completos, derivando en 16 *features* con 503 ejemplos. Este *dataset* mantiene todos los *features* disponibles (datos clínicos y de laboratorio) a costa de perder cantidad de registros. Las tablas (Tabla 3.8 y Tabla 3.9) resumen los *features* contenidos en este conjunto de datos.
- `Dataset Clínica+Glucemia basal (DCG)`. Conjunto de datos que mantiene la información clínica disponible y el único *feature* de laboratorio que no posee valores nulos (`glucemia_basal`): el resto de las variables de laboratorio fueron eliminadas. De esta forma, este *dataset* cuenta con 10 *features* y 1233 ejemplos. En contraposición con `DCL` aquí se preserva la cantidad de registros frente al valor aportado por el resto de los *features* de laboratorio. Las tablas (Tabla 3.10 y Tabla 3.11) resumen los *features* contenidos en este conjunto de datos.

Las versiones binarias de los conjuntos de datos son `DCL-bin` y `DCG-bin` respectivamente, los cuales traducen el problema a uno de clasificación binaria (“Sin riesgo” o “Con riesgo”).

Tabla 3.8: *Features* categóricos asociados a `DCL`/`DCL-bin`. A fin de evitar duplicidad de información, el *feature* de `clase` se agrega como una fila adicional para `DCL-bin`.

Feature	Valor	N	Porcentaje
sexo	0 ⇒ Femenino (f)	321	64%
	1 ⇒ Masculino (m)	182	36%
rango_edad	0 ⇒ menor de 45 años	12	2%
	1 ⇒ entre 45 y 54 años	201	40%
	2 ⇒ entre 55 y 64 años	206	41%
	3 ⇒ mayor de 64 años	84	17%
rango_imc	0 ⇒ menor de 25 IMC	5	1%
	1 ⇒ entre 25 y 30 IMC	87	17%
	2 ⇒ mayor de 30 IMC	411	82%
rango___cintura	0 ⇒ m: menos de 94cm	2	0.3%
	0 ⇒ f: menos de 80cm		
	1 ⇒ m: entre 94cm y 102cm	85	17%
	1 ⇒ f: entre 80cm y 88 cm		
	2 ⇒ m: mayor de 102 cm	416	82,7%
	2 ⇒ f: mayor de 88cm		
actividad_fisica	0 ⇒ No	470	93%
	1 ⇒ Sí	33	7%
cons___hortalizas	0 ⇒ No todos los días	438	87%
	1 ⇒ Todos los días	65	13%
toma___hta	0 ⇒ No	82	16%
	1 ⇒ Sí	421	84%
le___hiperglucemia	0 ⇒ No	457	91%
	1 ⇒ Sí	46	9%
le_diag___familiar	0 ⇒ No	81	16%
	1 ⇒ Sí: abuelo, tío, tía, o primo	311	62%
	2 ⇒ Sí: padre, hermano/a e hijo/a	111	22%
clase (`DCL`)	0 ⇒ Normal	229	46%
	1 ⇒ PDM (GAA, TGA o GAA+TGA)	251	50%
	2 ⇒ DM	23	5%
clase (`DCL-bin`)	0 ⇒ Sin riesgo	229	46%
	1 ⇒ Con riesgo	274	54%

Tabla 3.9: *Features* continuos asociados a DCL/DCL-bin.

Feature	Media	Desviación estándar	Mínimo	Máximo
glucemia_basal	102,274	12,394	45	143
colesterol_ldl	120,272	36,623	36	265
colesterol_total	198,149	41,373	87	369
colesterol_hdl	50,189	14,470	15	189
trigliceridos	142,753	68,567	37	519
creatinina_basal	0,794	0,219	0,21	2,57
hem___glucosilada	5,633	0,433	4,1	7,4

Tabla 3.10: *Features* categóricos asociados a DCG/DCG-bin. A fin de evitar duplicidad de información, el *feature* de clase se agrega como una fila adicional para DCG-bin.

Feature	Valor	N	Porcentaje
sexo	0 ⇒ Femenino (f)	839	68%
	1 ⇒ Masculino (m)	394	32%
rango_edad	0 ⇒ menor de 45 años	205	17%
	1 ⇒ entre 45 y 54 años	435	35%
	2 ⇒ entre 55 y 64 años	428	35%
	3 ⇒ mayor de 64 años	165	13%
rango_imc	0 ⇒ menor de 25 IMC	93	8%
	1 ⇒ entre 25 y 30 IMC	318	26%
	2 ⇒ mayor de 30 IMC	822	67%
rango___cintura	0 ⇒ m: menos de 94cm	55	4%
	0 ⇒ f: menos de 80cm		
	1 ⇒ m: entre 94cm y 102cm	205	17%
	1 ⇒ f: entre 80cm y 88 cm		
	2 ⇒ m: mayor de 102 cm	973	79%
rango___cintura	2 ⇒ f: mayor de 88cm	973	79%
actividad_fisica	0 ⇒ No	912	74%
	1 ⇒ Sí	321	26%
cons___hortalizas	0 ⇒ No todos los días	819	66%
	1 ⇒ Todos los días	414	34%
toma___hta	0 ⇒ No	497	40%
	1 ⇒ Sí	736	60%
le___hiperglucemia	0 ⇒ No	999	81%
	1 ⇒ Sí	234	19%

Continúa en la siguiente página

Tabla 3.10 – continuación de la página anterior

Feature	Valor	N	Porcentaje
le_diag___familiar	0 ⇒ No	395	32%
	1 ⇒ Sí: abuelo, tío, tía, o primo	412	33%
	2 ⇒ Sí: padre, hermano/a e hijo/a	426	35%
clase (DCG)	0 ⇒ Normal	620	50%
	1 ⇒ PDM (GAA, TGA o GAA+TGA)	479	39%
	2 ⇒ DM	134	11%
clase (DCG-bin)	0 ⇒ Sin riesgo	620	50%
	1 ⇒ Con riesgo	613	50%

Tabla 3.11: *Features* continuos asociados a DCG/DCG-bin.

Feature	Media	Desviación estándar	Mínimo	Máximo
glucemia_basal	104,093	26,396	45	482

3.2.2 *Datasets* con información clínica

En segundo lugar, se planteó sólo considerar los datos clínicos del paciente de forma de contar con un modelo más sencillo, sin costo y factible de realizar en cualquier momento, más allá del impacto que pudiera tener la eliminación de los datos de laboratorio en el rendimiento. Aquí, se genera un único conjunto de datos denominado `Dataset Clínica (DC)`, donde se tienen 9 *features* y 1233 ejemplos. Alternativamente, se cuenta con una versión binaria (`DC-bin`). La Tabla 3.12 resume los *features* contenidos en este conjunto de datos.

Tabla 3.12: *Features* categóricos asociados a DC/DC-bin. A fin de evitar duplicidad de información, el *feature* de clase se agrega como una fila adicional para DC-bin.

Feature	Valor	N	Porcentaje
sexo	0 ⇒ Femenino (f)	839	68%
	1 ⇒ Masculino (m)	394	32%
rango_edad	0 ⇒ menor de 45 años	205	17%
	1 ⇒ entre 45 y 54 años	435	35%
	2 ⇒ entre 55 y 64 años	428	35%
	3 ⇒ mayor de 64 años	165	13%
rango_imc	0 ⇒ menor de 25 IMC	93	8%
	1 ⇒ entre 25 y 30 IMC	318	26%
	2 ⇒ mayor de 30 IMC	822	67%
rango___cintura	0 ⇒ m: menos de 94cm	55	4%
	0 ⇒ f: menos de 80cm		
	1 ⇒ m: entre 94cm y 102cm	205	17%
	1 ⇒ f: entre 80cm y 88 cm		
	2 ⇒ m: mayor de 102 cm	973	79%
	2 ⇒ f: mayor de 88cm		
actividad_fisica	0 ⇒ No	912	74%
	1 ⇒ Sí	321	26%
cons___hortalizas	0 ⇒ No todos los días	819	66%
	1 ⇒ Todos los días	414	34%
toma___hta	0 ⇒ No	497	40%
	1 ⇒ Sí	736	60%
le___hiperglucemia	0 ⇒ No	999	81%
	1 ⇒ Sí	234	19%
le_diag___familiar	0 ⇒ No	395	32%
	1 ⇒ Sí: abuelo, tío, tía, o primo	412	33%
	2 ⇒ Sí: padre, hermano/a e hijo/a	426	35%
clase (DC)	0 ⇒ Normal	620	50%
	1 ⇒ PDM (GAA, TGA o GAA+TGA)	479	39%
	2 ⇒ DM	134	11%
clase (DC-bin)	0 ⇒ Sin riesgo	620	50%
	1 ⇒ Con riesgo	613	50%

3.2.3 *Datasets* con información de laboratorio

Por último, a modo de contraposición a la segmentación previa, se plantea la posibilidad de trabajar únicamente con la información de laboratorio. Aunque su viabilidad práctica puede ser limitada debido a la falta de disponibilidad de datos de laboratorio actualizados en todas las personas, los resultados obtenidos mediante este conjunto de datos

pueden complementar aquellos obtenidos mediante los anteriores. En este contexto, se ha generado un único conjunto de datos denominado `Dataset Laboratorio (DL)`, que consta de 7 *features* y 503 ejemplos. Al igual que los conjuntos anteriores, su versión binaria se denota como `DL-bin`. Las Tablas 3.13 y 3.14 resumen los *features* contenidos en este conjunto de datos.

Tabla 3.13: *Features* categóricos asociados a `DL/DL-bin`. A fin de evitar duplicidad de información, el *feature* de `clase` se agrega como una fila adicional para `DL-bin`.

Feature	Valor	N	Porcentaje
clase (DL)	0 ⇒ Normal	229	46%
	1 ⇒ PDM (GAA, TGA o GAA+TGA)	251	50%
	2 ⇒ DM	23	5%
clase (DL-bin)	0 ⇒ Sin riesgo	229	46%
	1 ⇒ Con riesgo	274	54%

Tabla 3.14: *Features* continuos asociados a `DL/DL-bin`.

Feature	Media	Desviación estándar	Mínimo	Máximo
glucemia_basal	102,274	12,394	45	143
colesterol_ldl	120,272	36,623	36	265
colesterol_total	198,149	41,373	87	369
colesterol_hdl	50,189	14,470	15	189
trigliceridos	142,753	68,567	37	519
creatinina_basal	0,794	0,219	0,21	2,57
hem___glucosilada	5,633	0,433	4,1	7,4

3.3 Alcances y limitaciones

Resulta importante aclarar que los modelos predictivos a desarrollar no pretenden reemplazar las pruebas PTOG como mecanismo de diagnóstico de DM y PDM. Como se explicó anteriormente, la DT2 es una enfermedad de difícil detección debido a la ausencia de síntomas específicos y/o falta de conocimiento de los factores de riesgo asociados. En ese sentido, los modelos predictivos propuestos buscan identificar aquellas personas de la población Argentina que tengan alta probabilidad de padecer DM no diagnosticada o PDM y desconozcan su condición. Para confirmar el diagnóstico, las personas con alta probabilidad, arrojada por el modelo, deberán realizar eventualmente una PTOG. Los modelos ayudarían a identificar quienes deben realizarlo y suplirían la ausencia de herramientas de este tipo siempre con la premisa de una detección temprana para no llegar a un punto de la enfermedad irreversible.

Asimismo, es importante remarcar que en el mejor de los casos se cuenta con un conjunto de datos con 1233 registros, lo cual es relativamente chico imponiendo las limitaciones típicas de un *dataset* reducido.

La segmentación que sólo utiliza la glucemia basal junto con la información clínica del paciente responde a su correlación con la clase a predecir, siendo uno de los valores que definen el resultado de la PTOG. Frente a esto, quizás el resto de la información de laboratorio sea redundante, así como también, sólo contar con información clínica sea insuficiente.

Inicialmente, la aproximación de analizar únicamente los datos de laboratorio, dada por las segmentaciones `DL` y `DL-bin`, fue considerada con el objetivo de garantizar la "completitud" en la división de datos entre información clínica y de laboratorio. No obstante, desde una perspectiva médica, se ha llegado a la conclusión de que esta aproximación carece de sentido. La información clínica proporciona una perspectiva integral de la condición del paciente, y limitarse únicamente a los datos de laboratorio impide una evaluación completa y precisa. Por consiguiente, se descarta esta segmentación en los experimentos a realizar, dado que no aporta valor clínico significativo.

Aunque el proceso de binarización (ver Sección 3.1.2.4) pueda favorecer al equilibrio de clases y la simplificación del problema, el costo radica en no poder diferenciar los casos de PDM de aquellos con DM. Sin embargo, desde el punto de vista médico, esto no sería tan grave, ya que lo importante es identificar pacientes en riesgo (sin importar su grado).

Las distintas segmentaciones propuestas intentan dilucidar la compensación entre la utilización de todos los *features* disponibles frente a reducir la cantidad de registros, o bien, reducir el número de variables con el objetivo de aumentar los ejemplos.

Tanto los modelos predictivos a evaluar como las distintas segmentaciones generan un conjunto de posibilidades que, a priori, abren un abanico para cubrir distintas necesidades a futuro así como también hacen más nutritiva y completa las comparaciones abordadas en la presente investigación.

CAPÍTULO 4

RESULTADOS EXPERIMENTALES

Cuando un *dataset* posee muchos atributos, puede resultar difícil de entender, visualizar y procesar. Por eso, el presente capítulo iniciará con una introducción y análisis de reducción de dimensionalidad para `DCL-bin` a fin de obtener una mejor visualización de la complejidad del problema a resolver (Sección 4.1). Posteriormente, se experimentarán diversos modelos de clasificación en cada una de las segmentaciones que abordan dicho problema, describiendo así los resultados obtenidos (Secciones 4.2, 4.3 y 4.4). Con el objetivo de identificar atributos relevantes, se llevará a cabo un proceso de selección de *features* a partir de `DCL-bin` (Sección 4.5). Luego, se realizará una experimentación sobre `DCL` aplicando diferentes modelos de regresión junto con su evaluación correspondiente (Sección 4.6). Finalmente, se efectuará un análisis y comparación general respecto a los resultados obtenidos (Sección 4.7).

4.1 Introducción a las experimentaciones

El presente capítulo se centra en el desarrollo y evaluación de modelos para las distintas segmentaciones anteriormente planteadas. Para ello, se ha llevado a cabo un proceso de exploración, preprocesamiento, entrenamiento y evaluación de diversos algoritmos utilizando *scikit-learn*. Asimismo, se ha utilizado *TensorFlow* para la construcción de modelos de ANN. Para la manipulación, análisis y visualización de los datos, se han utilizado *Pandas*, *NumPy* y *Matplotlib*, lo que ha permitido una gestión efectiva y eficiente de estos.

Con el objetivo de centralizar, modularizar y reutilizar el precargado de las distintas segmentaciones planteadas y disponibilizar ciertas utilidades, se ha desarrollado un módulo *Python* llamado *diabetes*. El mismo posee ciertas funciones cuyo nombre identifica la segmentación a precargar y ciertos parámetros a continuación descriptos:

- *binary : bool*. Parámetro por el cual puede obtenerse la versión para clasificación binaria del *dataset*. Si es `True` se obtiene la versión de clasificación binaria. Caso contrario, si es `False`, se obtiene la versión multiclase.

- *return_X_y : bool, optional*. Parámetro opcional que permite obtener el *dataset* como una tupla de dos elementos. Si es `True`, se retorna una tupla donde el primer elemento es una matriz *NumPy* que contiene los datos de entrada del *dataset*, y el segundo elemento es una matriz *NumPy* que contiene las etiquetas de clase o variables de salida correspondientes a cada muestra en los datos de entrada. Por defecto, su valor es `False`, en cuyo caso se obtiene un objeto *Bunch*.

Tomando como base a *scikit-learn*, quien utiliza el objeto *Bunch* para la carga de ciertos *datasets*[1], las funciones de *diabetes* retornan un *Bunch* que agrupa los siguientes atributos:

- *data*. Matriz *Numpy* utilizada para el entrenamiento y evaluación de los modelos. Su forma varia dependiendo el conjunto de datos precargado con (x, y), donde x representa la cantidad de ejemplos e y la cantidad de *features*.

- *target*. Matriz *Numpy* utilizada para el entrenamiento y evaluación de los modelos, la cual representa las etiquetas de clase o variables de salida de la matriz anterior (*data*).

- *feature_names*. Lista de nombres de *features* o columnas del conjunto de datos.

- *target_names*. Lista de nombres de *target* o clases del conjunto de datos.

- *frame*. *DataFrame*[2] que encapsula la *data* y *target* del conjunto de datos.

[1] Como el conjunto de datos `Iris`, donde se puede cargar utilizando la función `load_iris()`. La misma retorna un objeto *Bunch* que contiene los datos del *dataset* `Iris` y su información descriptiva.

[2] Estructura de datos bidimensional de *Pandas* que agrupa filas y columnas, donde cada columna puede contener datos de diferentes tipos.

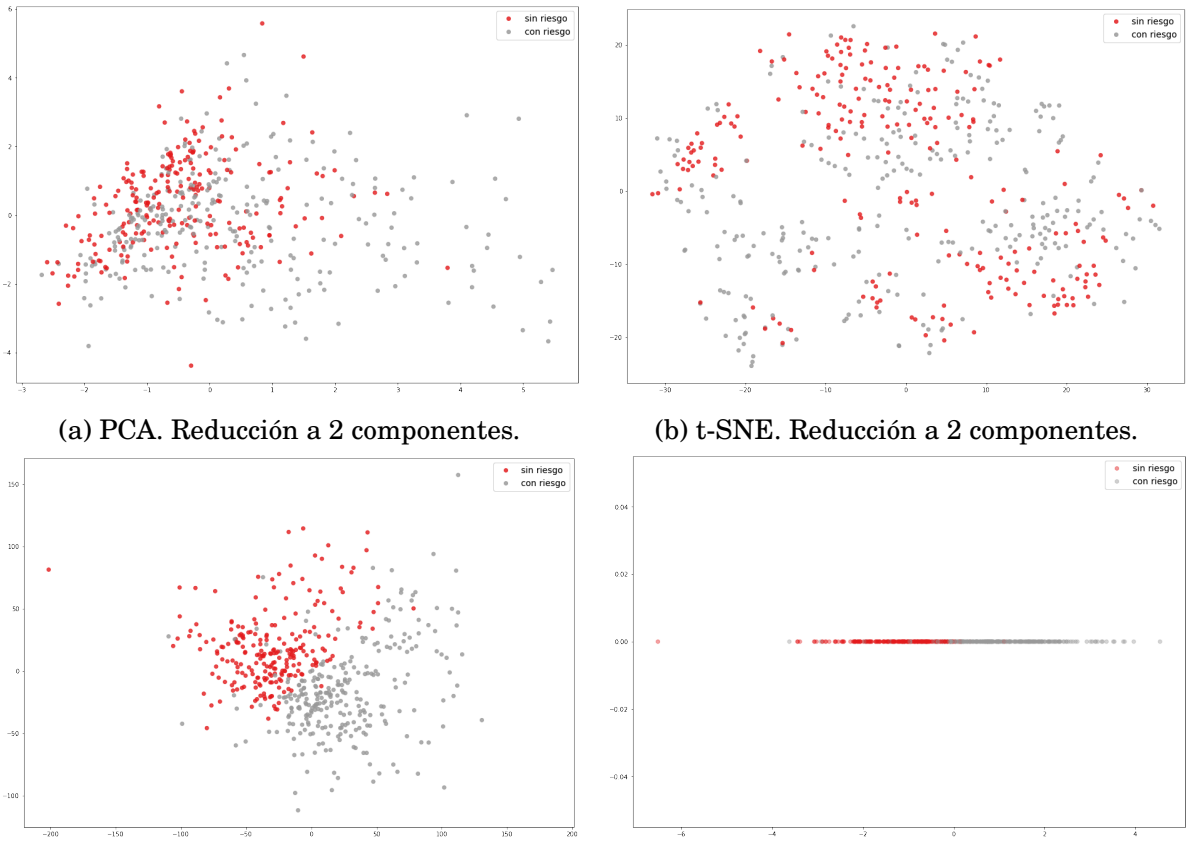
- *filename*. Ruta a la localización del conjunto de datos.

De acuerdo con los lineamientos habituales en el área, se procedió a realizar una partición de las distintas segmentaciones en dos conjuntos: el conjunto de entrenamiento y el conjunto de evaluación. Adicionalmente, con el fin de minimizar posibles sesgos en el análisis, se utilizó una técnica CV con muestreo estratificado aleatorio (*StratifiedShuffleSplit* [152]). Este enfoque difiere de una simple división aleatoria de los datos, ya que garantiza que la proporción de cada clase en el conjunto de entrenamiento y prueba sea idéntica a la proporción existente en el conjunto original de datos. Esto resulta especialmente útil en situaciones en las que el conjunto de datos presenta un desequilibrio en su distribución de clases. En el caso particular de esta investigación, se especificó un valor de $n\_splits = 50$, lo que indica la cantidad de veces que se llevará a cabo la partición de los datos en conjuntos de entrenamiento y evaluación. Además, se reservó un 30% del *dataset* para el conjunto de evaluación. Es importante destacar que, debido a que las variables de entrada poseen magnitudes diferentes, se aplicó una normalización *min-max* para todos los modelos excepto para las ANN donde se utilizó una normalización estándar.

Todas las experimentaciones se realizaron sobre una computadora local utilizando *Jupyter Notebook* [153]. El hardware disponible fue una CPU Intel Core i7 2,6 GHz Quad-Core con 16 GB de memoria RAM sobre la que se ejecuta un sistema operativo macOS.

Como se mencionó en la Sección 2.2.3.1, el proceso de reducción de dimensionalidad, bajo la utilización de extracción de *features*, permite transformar los datos de un espacio de alta dimensión en un espacio más acotado. Al reducir la cantidad de *features* en un conjunto de datos, es posible simplificar la complejidad del problema y hacer que sea más fácil de entender, analizar y visualizar.

Partiendo de la premisa anteriormente mencionada, se exploraron diversos algoritmos de reducción de dimensionalidad en una de las segmentaciones planteadas (`DCL-bin`). La Figura 4.1 ofrece una representación visual más clara de la complejidad del problema de clasificación al transformar el *dataset* de 16 *features* a un conjunto de datos en dos dimensiones. En particular, se probaron distintos algoritmos de reducción de dimensionalidad, tanto supervisados (NCA y LDA) como no supervisados (PCA y t-SNE), con el objetivo de obtener diferentes representaciones y analizar en profundidad la naturaleza del problema. En todos los casos, se logró una reducción de la dimensionalidad a 2 componentes, a excepción de LDA, donde se impone la restricción $n\_componentes < n\_classes - 1$, resultando en una proyección lineal en lugar de un plano bidimensional.



(a) PCA. Reducción a 2 componentes.

(b) t-SNE. Reducción a 2 componentes.

(c) NCA. Reducción a 2 componentes.

(d) LDA. Reducción a 1 componente.

Figura 4.1: Algoritmos de reducción de dimensionalidad aplicados sobre `DCL-bin`.

Como se observa en la Figura 4.1, la reducción realizada con NCA (4.1c) parece ser la opción más adecuada, ya que las clases de los datos pueden ser claramente diferenciadas. Sin embargo, los resultados obtenidos con los demás algoritmos sugieren que el problema es más complejo.

4.2 Modelos de clasificación para `DCL-bin`

Con el objetivo de obtener una aproximación inicial, se llevó a cabo un análisis de `DCL-bin` mediante la utilización de un modelo LOR con un enfoque en el balanceo de clases (`class_weight=‘‘balanced’’`[3]). Se reservó un porcentaje del 30% de `DCL-bin` para la evaluación, empleando la técnica CV descrita en la Sección 4.1 y aplicando una normalización *min-max*. A continuación, la Tabla 4.1 agrupa las métricas obtenidas en ambos conjuntos:

[3]Utilizado para tratar el problema de clases desequilibradas. El parámetro `balanced` permitirá ajustar automáticamente los pesos según el número de muestras en cada clase.

Tabla 4.1: Métricas del modelo LOR para DCL-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	87.73% ± 1.38%	83.48% ± 2.01%	92.46% ± 1.40%	$\overline{160}$
Con riesgo	88.69% ± 1.42%	93.11% ± 1.22%	84.71% ± 2.16%	$\overline{192}$
Accuracy: 88.23% ± 1.40%				
Evaluación				
Sin riesgo	85.05% ± 3.17%	81.01% ± 4.24%	89.68% ± 3.63%	$\overline{69}$
Con riesgo	86.01% ± 3.30%	90.50% ± 3.11%	82.10% ± 4.81%	$\overline{82}$
Accuracy: 85.56% ± 3.21%				

La Tabla 4.1 muestra los resultados del modelo LOR para ambos conjuntos en la media del total de iteraciones (50). En el conjunto de entrenamiento, el modelo LOR obtuvo un *accuracy* del 88.23%, con desviaciones inferiores al 1.5%. Por otro lado, en el conjunto de evaluación, dicha métrica alcanzó un 85.56%, con desviaciones menores al 3.5%. Estos resultados sugieren que el modelo tiene una capacidad aceptable para generalizar a nuevos datos, clasificando correctamente aproximadamente 8 de cada 10 muestras. Además, la diferencia en el rendimiento del modelo entre los conjuntos de entrenamiento y evaluación indica que el modelo presenta una variación razonablemente baja en su rendimiento.

La Figura 4.2 muestra la matriz de confusión del modelo LOR para DCL-bin sobre el conjunto de evaluación. Se observa que el modelo clasificó correctamente 128 muestras y tuvo 21 errores de clasificación. De los errores, el modelo presentó más falsos negativos (14) que falsos positivos (7), lo que sugiere que el modelo es capaz de identificar correctamente la mayoría de los casos de la clase “Sin riesgo”, pero tiene más dificultades para identificar correctamente los casos de la clase “Con riesgo”.

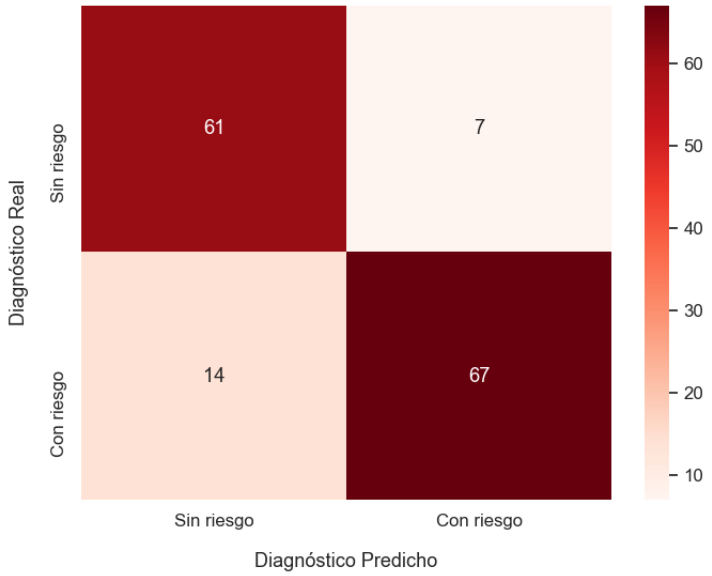


Figura 4.2: Matriz de confusión del modelo LOR sobre el conjunto de evaluación para `DCL-bin`.

En cuanto a la métrica de *precision*, el modelo LOR presenta porcentajes más altos para la clase "Con riesgo", alcanzando un 93.11% en entrenamiento y un 90.50% en evaluación. Esto significa que más de 9 de cada 10 personas a las que se les dice que tienen riesgo de diabetes, realmente lo tienen.

Por otra parte, el modelo LOR, presenta un valor de *recall* ligeramente menor para la clase "Con riesgo" que para la clase "Sin riesgo" tanto en entrenamiento como en evaluación. En evaluación presenta el valor más bajo de todas las métricas con un 82.10%. Esto sugiere que de aproximadamente 2 de cada 10 pacientes que tienen riesgo no son identificados por el modelo.

No obstante, los valores de *F1-score* son razonablemente altos en ambos conjuntos. Esto indica que el modelo está funcionando de manera efectiva para ambas clases.

Para concluir con el análisis del modelo LOR aplicado a `DCL-bin`, se presenta en la Figura 4.3 los pesos asignados a cada atributo. Como es posible observar, la mayoría carece de relevancia para la clasificación. No obstante, se identifica una lógica discernible en la importancia relativa de dos características específicas: la `glucemia_basal` se destaca como el atributo más influyente en la predicción del riesgo de diabetes, mientras que la `hemoglobina_glucosilada` muestra un impacto de menor predominancia.

En relación a otros atributos, resulta llamativo el impacto negativo que tienen la `actividad_fisica` y `consume_vegetales_frutas_y_hortalizas` en la clasificación. Sin embargo, se ha identificado un desequilibrio en los datos correspondientes a ambos atributos, lo cual afecta la asignación de los coeficientes por parte del modelo.

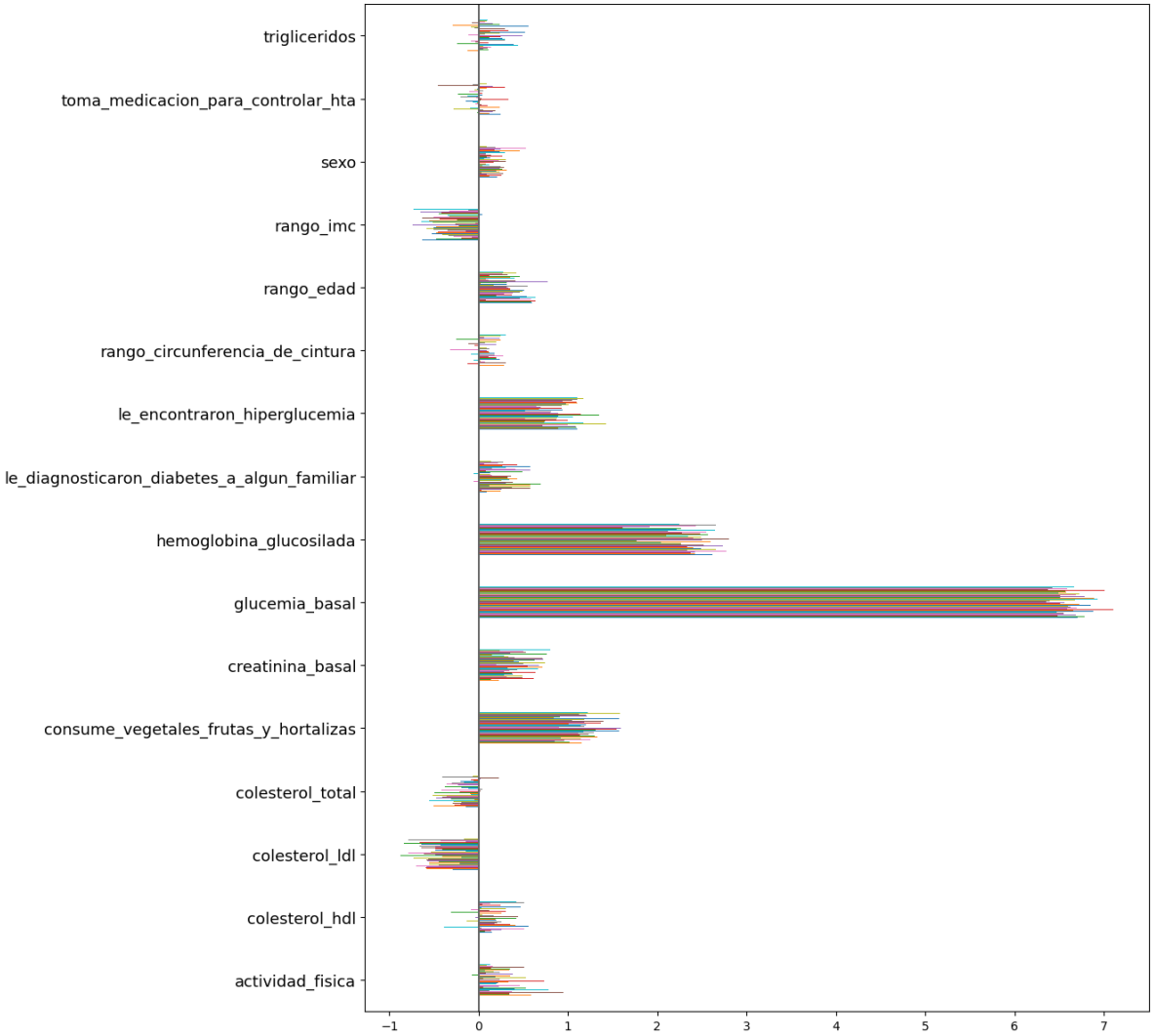
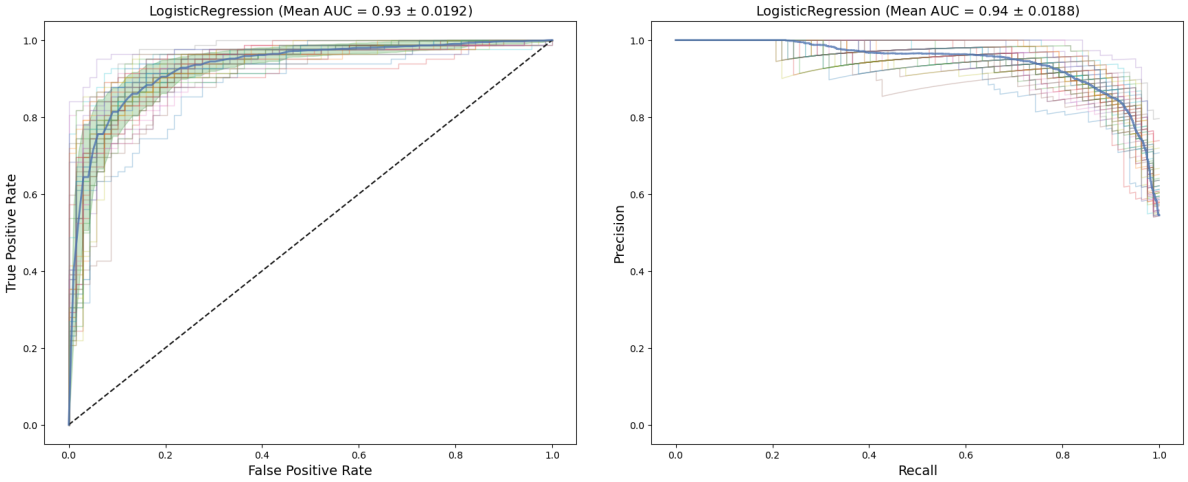


Figura 4.3: Pesos asociados a cada *feature* por el modelo LOR sobre el conjunto de evaluación para DCL-bin. Cada una de las ejecuciones de CV se encuentra representada por distintas líneas de colores.

A continuación, se presenta en la Figura 4.4 las curvas ROC y PR generadas a partir del conjunto de evaluación para `DCL-bin`. El modelo obtenido posee una media `AUC = 0.93` y una media `AUC = 0.94` para las curvas ROC y PR respectivamente. Esto evidencia un rendimiento satisfactorio durante la clasificación de muestras. El valor de `AUC` para la curva ROC indica que el modelo logra distinguir entre las dos clases en una media del 93% de los casos. Por otro lado, el `AUC` para la curva PR sugiere que el modelo logra un buen balance entre *precision* y *recall* independientemente del umbral de clasificación utilizado.



(a) ROC (b) PR

Figura 4.4: Curvas ROC y PR del modelo LOR sobre el conjunto de evaluación para `DCL-bin`.

A fines comparativos, se exponen a continuación las métricas correspondientes a los modelos restantes, los cuales fueron entrenados y evaluados de la misma forma que el modelo LOR, esto es, reservando un porcentaje del 30% de `DCL-bin` para la evaluación, empleando la técnica CV y aplicando una normalización *min-max*.

De esta forma, se definió un modelo DT con una profundidad máxima de 5 y un balanceo de clases (`max_depth = 5, class_weight='balanced'`). En la Tabla 4.2 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 97.93% y 93.46% para los conjuntos de entrenamiento y evaluación respectivamente. Es relevante notar que las desviaciones observadas en estas métricas son mínimas, indicando una consistencia en el rendimiento del modelo.

Este desempeño representa una mejora significativa con respecto al modelo previamente discutido (LOR). Adicionalmente, las métricas de *precision* y *recall* exhiben un alto nivel de efectividad en la identificación de ambas clases. Aproximadamente 9 de cada 10 personas a las que el modelo predice riesgo de diabetes, realmente la padece. Además, se presenta un alto valor de *F1-score* indicando un buen equilibrio entre *precision* y *recall* para ambas clases.

Con el propósito de brindar una visión más completa del desempeño del modelo, se presentan en las Figuras 4.5 y 4.6 la matriz de confusión y las curvas ROC y PR,

respectivamente. Estas se basan en las predicciones generadas por el modelo en el conjunto de evaluación correspondiente a DCL-bin. La matriz de confusión (Figura 4.5) exhibe el rendimiento del modelo con una reducida cantidad de errores, tan solo 9, mientras que logra clasificar correctamente un total de 140 muestras. Las curvas (Figura 4.6), a su vez, reflejan y reafirman este desempeño destacado.

Tabla 4.2: Métricas del modelo DT para DCL-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	97.77% ± 0.59%	95.73% ± 1.21%	99.91% ± 0.31%	$\overline{160}$
Con riesgo	98.06% ± 0.53%	99.93% ± 0.26%	96.27% ± 1.09%	$\overline{192}$
Accuracy: 97.93% ± 0.56%				
Evaluación				
Sin riesgo	93.01% ± 1.69%	90.89% ± 2.50%	95.33% ± 2.71%	$\overline{69}$
Con riesgo	93.84% ± 1.50%	95.97% ± 2.24%	91.88% ± 2.49%	$\overline{82}$
Accuracy: 93.46% ± 1.58%				

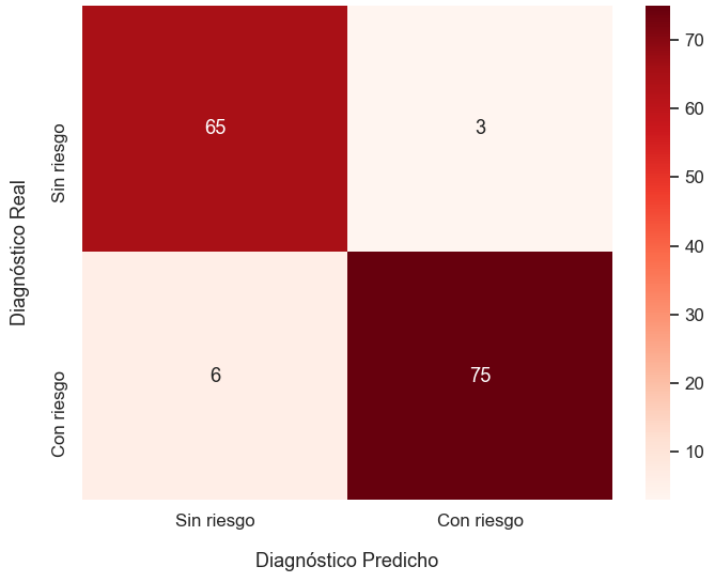
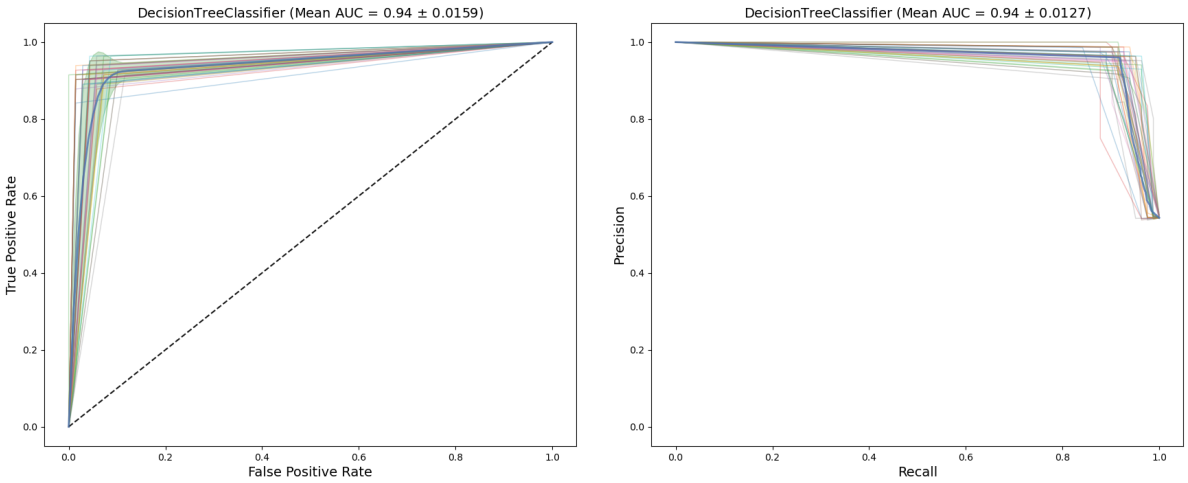


Figura 4.5: Matriz de confusión del modelo DT sobre el conjunto de evaluación para DCL-bin.



(a) ROC (b) PR

Figura 4.6: Curvas ROC y PR del modelo DT sobre el conjunto de evaluación para `DCL-bin`.

Posteriormente, se definió un modelo k-NN con 7 vecinos (`n_neighbors = 7`). En la Tabla 4.3 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 81.37% y 71.62% para los conjuntos de entrenamiento y evaluación respectivamente. Como es posible observar, hasta el momento, este modelo registra el rendimiento más bajo en términos de *accuracy*, en comparación con los modelos previamente considerados.

A pesar de esto, las métricas generales del modelo k-NN tienen una desviación similar a las observadas en el modelo LOR, aunque con un decremento del 10% en los valores de entrenamiento y evaluación.

Las Figuras 4.7 y 4.8 exhiben la matriz de confusión y las curvas ROC y PR, generadas a partir del conjunto de evaluación correspondiente a `DCL-bin`, respectivamente. Como era de anticiparse, la matriz de confusión (Figura 4.7) muestra un desempeño inferior en comparación con los modelos presentados anteriormente, evidenciando un mayor número de errores (41). Adicionalmente, las curvas (Figura 4.8) pueden notarse inferiores, en su comportamiento, a las analizadas anteriormente.

Tabla 4.3: Métricas del modelo k-NN para DCL-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	80.08% ± 1.92%	78.01% ± 2.66%	82.33% ± 2.35%	$\overline{160}$
Con riesgo	82.49% ± 1.96%	84.57% ± 1.81%	80.57% ± 2.99%	$\overline{192}$
Accuracy: 81.37% ± 1.92%				
Evaluación				
Sin riesgo	70% ± 4.06%	67.86% ± 4.48%	72.55% ± 5.63%	$\overline{69}$
Con riesgo	72.97% ± 4%	75.53% ± 3.87%	70.83% ± 5.75%	$\overline{82}$
Accuracy: 71.62% ± 3.82%				

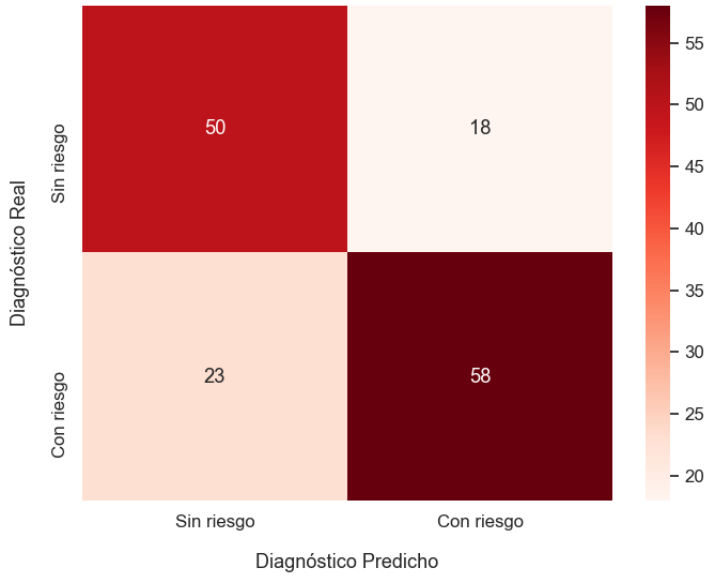
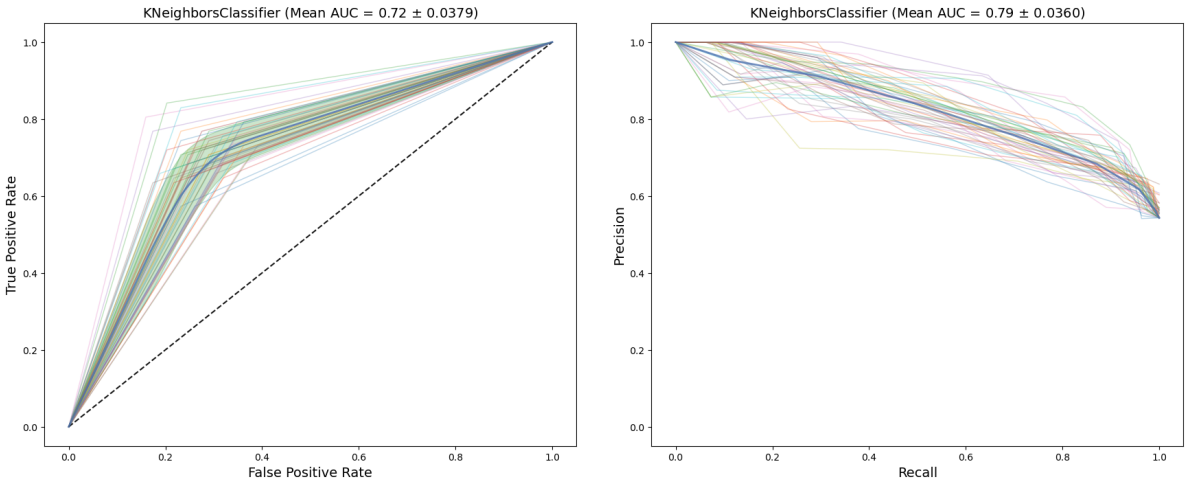


Figura 4.7: Matriz de confusión del modelo k-NN sobre el conjunto de evaluación para DCL-bin.



(a) ROC (b) PR

Figura 4.8: Curvas ROC y PR del modelo k-NN sobre el conjunto de evaluación para `DCL-bin`.

El modelo RF, configurado con el parámetro `max_depth = 2`, impone una restricción en la profundidad máxima de cada árbol dentro del bosque, limitándola a dos niveles. De esta manera, el modelo RF fue entrenado y se resumen los resultados obtenidos en la Tabla 4.4. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 95.95% en el conjunto de entrenamiento y del 94.56% en el conjunto de evaluación. Las desviaciones correspondientes a estas métricas son mínimas, similares a las del modelo DT.

Tanto el modelo RF como el DT demuestran un alto grado de efectividad en la clasificación de ambas clases, presentando un valor elevado de *F1-score*, lo que indica un buen equilibrio entre *precision* y *recall*. La Figura 4.10 muestra las curvas ROC y PR generadas a partir del conjunto de evaluación para DCL-bin. Ambas curvas se aproximan al comportamiento ideal, lo que evidencia el excelente rendimiento del modelo RF.

De igual forma, la Figura 4.9 presenta la matriz de confusión, la cual exhibe, hasta el momento, el mejor rendimiento con tan solo 7 errores para la clase “Con riesgo”, logrando clasificar correctamente un total de 142 muestras.

Tabla 4.4: Métricas del modelo RF para `DCL-bin`.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	95.73% ± 0.61%	92.01% ± 1.16%	99.79% ± 0.39%	$\overline{160}$
Con riesgo	96.15% ± 0.59%	99.81% ± 0.34%	92.76% ± 1.14%	$\overline{192}$
Accuracy: 95.95% ± 0.60%				

Continúa en la siguiente página

Tabla 4.4 – continuación de la página anterior

	F1-score	precision	recall	N
Evaluación				
Sin riesgo	94.32% ± 1.61%	90.46% ± 2.69%	98.58% ± 1.65%	$\overline{69}$
Con riesgo	94.77% ± 1.60%	98.73% ± 1.46%	91.17% ± 2.77%	$\overline{82}$
Accuracy: 94.56% ± 1.60%				

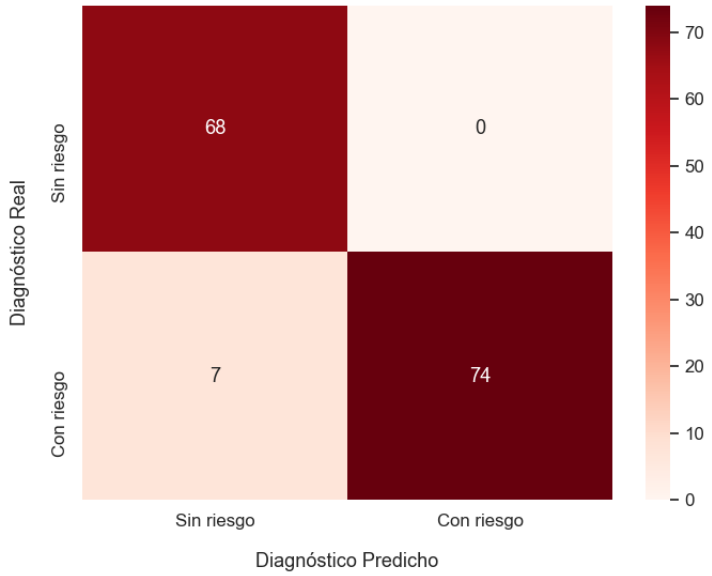
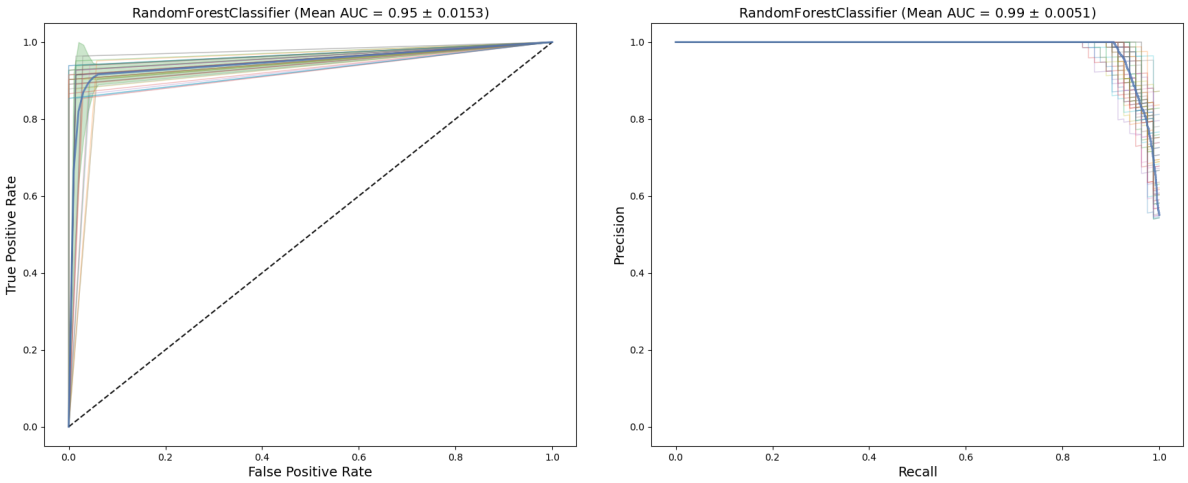


Figura 4.9: Matriz de confusión del modelo RF sobre el conjunto de evaluación para DCL-bin.



(a) ROC (b) PR

Figura 4.10: Curvas ROC y PR del modelo RF sobre el conjunto de evaluación para `DCL-bin`.

Con el objetivo de concluir el análisis de los modelos de clasificación para `DCL-bin`, se optó por experimentar la aplicabilidad de un modelo ANN. De esta forma, se construyó una ANN con una sola capa oculta. En dicha capa, se especificó un tamaño de 100 neuronas y se utilizó la función de activación `ReLU`. Asimismo, se aplicó una regularización L2 de 0.1 permitiendo controlar el sobreajuste del modelo. La capa de salida, posee una cantidad de neuronas igual al número de clases a clasificar (2), utilizando la función de activación sigmoidea. Esta capa produce las salidas del modelo, donde cada neurona representa la probabilidad de pertenecer a una de las clases, es decir, valores comprendidos entre 0 y 1.

Posteriormente, se procedió a compilar el modelo, utilizando un optimizador `Adamax` con una tasa de aprendizaje de 0.001. Como función de pérdida se utiliza la BCE, adecuada para problemas de clasificación binaria. Cabe destacar que las salidas generadas por el modelo ANN representan probabilidades en lugar de etiquetas de clases específicas.

Para evaluar el rendimiento del modelo, se recurrió a un umbral de 0.5, que se aplicó a las salidas de la red para su conversión en etiquetas de clase. En esencia, este proceso emplea la función $f(x)$ definida como:

$$f(x) = \begin{cases} 1 & \text{si } x > 0.5 \\ 0 & \text{caso contrario} \end{cases}$$

De acuerdo al modelo propuesto, en la Tabla 4.5 se presentan los resultados obtenidos con un entrenamiento bajo (`epochs = 60`, `batch_size = 16`) y una normalización estándar. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 97.17% en el conjunto de entrenamiento y del 91.22% en el conjunto de evaluación con desviaciones mínimas en ambos casos.

Con el propósito de brindar una visión más completa del desempeño del modelo,

se presentan en las Figuras 4.11 y 4.12 la matriz de confusión y las curvas ROC y PR, respectivamente. Estas se basan en las predicciones generadas por el modelo en el conjunto de evaluación correspondiente a DCL-bin. La matriz de confusión (Figura 4.11) exhibe un rendimiento similar a la del modelo DT. Las curvas (Figura 4.12), como en los modelos RF y DT, se aproximan al comportamiento ideal, lo que evidencia su muy buen rendimiento.

Tabla 4.5: Métricas del modelo ANN para DCL-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	96.93% ± 2.05%	95.69% ± 2.09%	98.20% ± 2.13%	$\overline{160}$
Con riesgo	97.38% ± 1.74%	98.47% ± 1.81%	96.31% ± 1.79%	$\overline{192}$
Accuracy: 97.17% ± 1.88%				
Evaluación				
Sin riesgo	90.44% ± 2.28%	89.96% ± 2.79%	91.04% ± 3.64%	$\overline{69}$
Con riesgo	91.87% ± 1.90%	92.47% ± 2.85%	91.37% ± 2.73%	$\overline{82}$
Accuracy: 91.22% ± 2.06%				

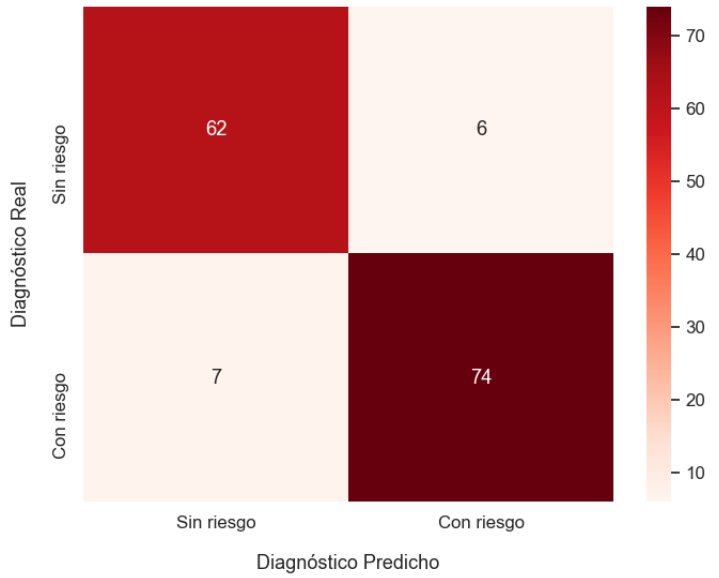
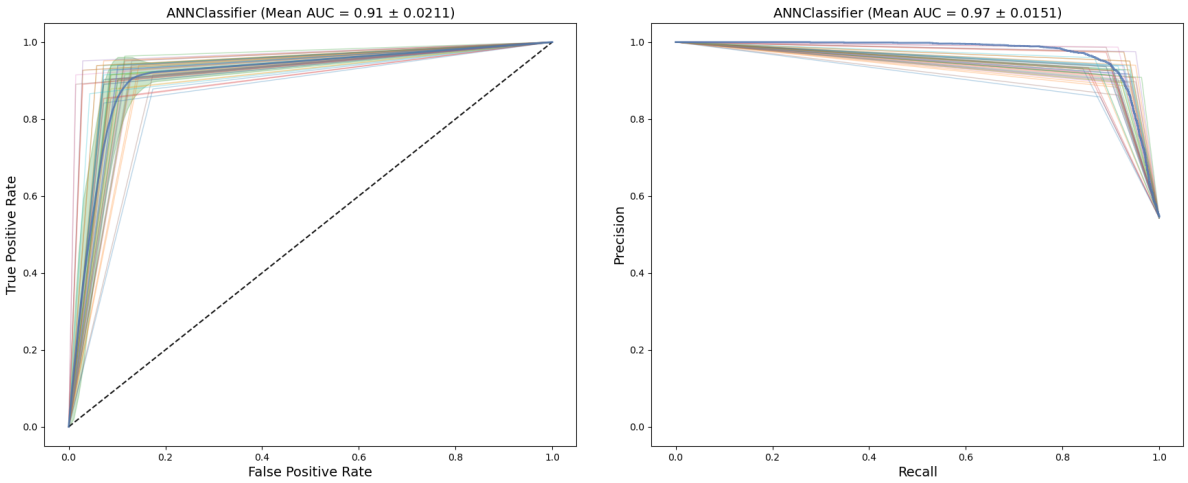


Figura 4.11: Matriz de confusión del modelo ANN sobre el conjunto de evaluación para DCL-bin.



(a) ROC (b) PR

Figura 4.12: Curvas ROC y PR del modelo ANN sobre el conjunto de evaluación para `DCL-bin`.

Finalmente, en la Tabla 4.6, se proporciona un resumen de las métricas obtenidas por los modelos experimentados en el conjunto de evaluación para `DCL-bin`. A diferencia de casos anteriores en los que se presentaban métricas para ambas clases, esta tabla resume las métricas para la clase "Con riesgo". Los modelos ANN, RF y DT exhiben un grado significativo de efectividad en la clasificación de ambas clases. Estos modelos presentan métricas muy similares entre sí. En contraste, el modelo LOR se ubica por debajo de ellos, seguido por el modelo k-NN, que muestra el menor porcentaje de *accuracy* con un 71.62%. El modelo RF, por otro lado, alcanza el mayor porcentaje de *accuracy* para `DCL-bin`, con un valor del 94.56%.

El porcentaje de *accuracy* obtenido para los modelos ANN, RF y DT indica su gran capacidad para generalizar hacia nuevos datos, clasificando correctamente aproximadamente 9 de cada 10 muestras. Asimismo, en términos de *precision*, todos los modelos experimentados muestran porcentajes más altos para la clase "Con riesgo". Esto indica que, en mayor o menor medida, los modelos que predicen pacientes con riesgo de diabetes realmente lo tienen. De manera similar, en cuanto a la métrica *recall*, los modelos siempre presentan porcentajes más altos para la clase "Sin riesgo", lo cual indica una baja tasa de pacientes no identificados con riesgo de diabetes.

Tabla 4.6: Resumen de los modelos experimentados para la clase "Con riesgo" en el conjunto de evaluación sobre DCL-bin.

Clasificador	accuracy	F1-score	precision	recall
RF (max_depth = 2)	94.56% ± 1.60%	94.77% ± 1.60%	98.73% ± 1.46%	91.17% ± 2.77%
DT (max_depth = 5, class_weight = ''balanced'')	93.46% ± 1.58%	93.84% ± 1.50%	95.97% ± 2.24%	91.88% ± 2.49%
ANN	91.22% ± 2.06%	91.87% ± 1.90%	92.47% ± 2.85%	91.37% ± 2.73%
LOR (class_weight = ''balanced'')	85.56% ± 3.21%	86.01% ± 3.30%	90.50% ± 3.11%	82.10% ± 4.81%
k-NN (n_neighbors = 7)	71.62% ± 3.82%	72.97% ± 4%	75.53% ± 3.87%	70.83% ± 5.75%

4.3 Modelos de clasificación para DCG-bin

En base a la destacada influencia de la variable glucemia_basal en la ponderación asignada por el modelo LOR para DCL-bin, se tomó la decisión de evaluar el efecto de incluir únicamente este atributo junto con la información clínica, lo cual resultó en la generación de la segmentación DCG.

Mediante el empleo del mismo enfoque utilizado para DCL-bin, se llevó a cabo un análisis de DCG-bin mediante la utilización de un modelo LOR con balanceo de clases (class_weight=''balanced''). De manera análoga, se reservó un porcentaje del 30% de DCG-bin para la evaluación, empleando la técnica CV descrita en la Sección 4.1 y aplicando una normalización *min-max*. A continuación, la Tabla 4.7 agrupa las métricas obtenidas en ambos conjuntos:

Tabla 4.7: Métricas del modelo LOR para `DCG-bin`.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	79.38% ± 2.66%	73.91% ± 3.79%	85.82% ± 1.97%	$\overline{434}$
Con riesgo	75.24% ± 4.12%	82.74% ± 2.50%	69.11% ± 5.63%	$\overline{429}$
Accuracy: 77.51% ± 3.26%				
Evaluación				
Sin riesgo	77.79% ± 3.73%	72.12% ± 4.53%	84.54% ± 3.62%	$\overline{186}$
Con riesgo	73.05% ± 5.30%	80.96% ± 4.44%	66.70% ± 6.55%	$\overline{184}$
Accuracy: 75.66% ± 4.37%				

La Tabla 4.7 muestra los resultados del modelo LOR para ambos conjuntos en la media del total de iteraciones (50). En el conjunto de entrenamiento, el modelo obtuvo un *accuracy* del 77.51%, con desviaciones inferiores al 3.5%. Por otro lado, en el conjunto de evaluación, dicha métrica alcanzó un 75.66%, con desviaciones menores al 4.5%. El modelo es capaz de clasificar correctamente aproximadamente 7 de cada 10 muestras.

La Figura 4.13 muestra la matriz de confusión del modelo LOR para `DCG-bin` sobre el conjunto de evaluación. Se observa que el modelo clasificó correctamente 279 muestras y tuvo 89 errores de clasificación. De los errores, el modelo presentó más falsos negativos (61) que falsos positivos (28), aunque, en este caso la proporción aumenta debido a un mayor tamaño del conjunto de datos.

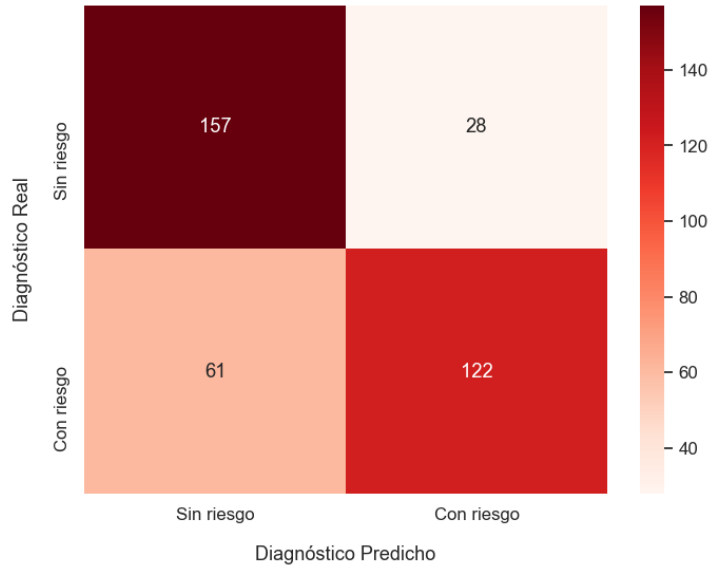


Figura 4.13: Matriz de confusión del modelo LOR sobre el conjunto de evaluación para DCG-bin.

En cuanto a la métrica de *precision*, el modelo LOR presenta porcentajes más altos para la clase "Con riesgo", alcanzando un 82.74% en entrenamiento y un 80.96% en evaluación. Esto significa que 8 de cada 10 personas a las que se les dice que tienen riesgo de diabetes, realmente lo tienen.

En términos de *recall*, el modelo LOR, presenta una diferencia marcada siendo mucho mayor para la clase "Sin riesgo" tanto en entrenamiento como en evaluación. Esto significa que el modelo es más efectivo en detectar pacientes sin riesgo de diabetes. Aproximadamente 4 de cada 10 pacientes que tienen riesgo no son identificados por el modelo.

Para finalizar el análisis del modelo LOR aplicado a DCG-bin, se presenta en la Figura 4.14 la distribución de pesos asignados a cada atributo. En ella se evidencia un patrón de asignación de pesos similar al analizado en la Figura 4.3. Se observa una clara predominancia de glucemia_basal, seguida por le_encontraron_hiper glucemia, aunque esta última tiene una influencia significativamente menor en la predicción del riesgo de diabetes.

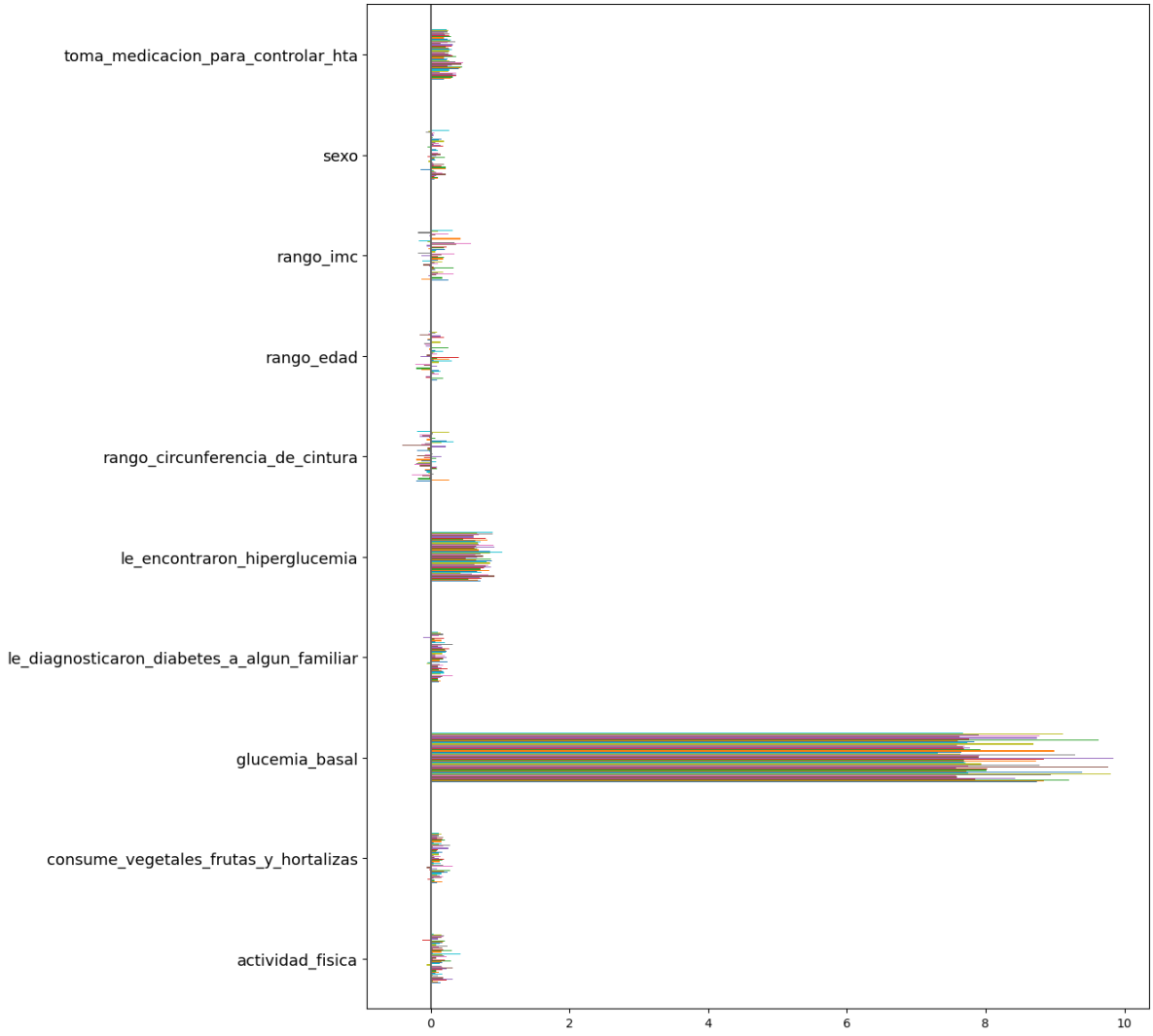
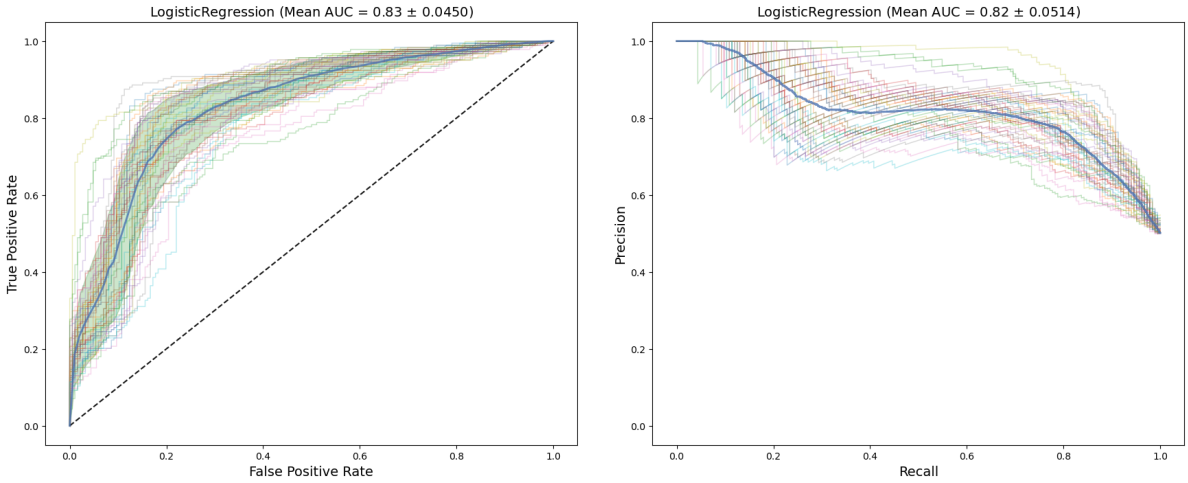


Figura 4.14: Pesos asociados a cada *feature* por el modelo LOR sobre el conjunto de evaluación para `DCG-bin`. Cada una de las ejecuciones de CV se encuentra representada por distintas líneas de colores.

A continuación, se presenta en la Figura 4.15 las curvas ROC y PR generadas a partir del conjunto de evaluación para DCG-bin. El modelo obtenido posee una media AUC = 0.83 y una media AUC = 0.82 para las curvas ROC y PR respectivamente.



(a) ROC (b) PR

Figura 4.15: Curvas ROC y PR del modelo LOR sobre el conjunto de evaluación para DCG-bin.

A fines comparativos, se exponen a continuación las métricas correspondientes a los modelos restantes, los cuales fueron entrenados y evaluados de la misma forma que el modelo LOR, esto es, reservando un porcentaje del 30% de DCG-bin para la evaluación, empleando la técnica CV y aplicando una normalización *min-max*.

De esta forma, se definió un modelo DT con una profundidad máxima de 5 y un balanceo de clases (max_depth = 5, class_weight=‘‘balanced’’). En la Tabla 4.8 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 93.70% y 91.87% para los conjuntos de entrenamiento y evaluación respectivamente. Es relevante notar que las desviaciones observadas en estas métricas son mínimas, indicando una consistencia en el rendimiento del modelo.

Este resultado representa una mejora significativa con respecto al modelo previamente discutido (LOR). Adicionalmente, las métricas de *precision* y *recall* exhiben un alto nivel de efectividad en la identificación de ambas clases. Aproximadamente 9 de cada 10 personas a las que el modelo predice riesgo de diabetes, realmente la padece. Además, se presenta un alto valor de *F1-score* indicando un buen equilibrio entre *precision* y *recall* para ambas clases.

Con el propósito de brindar una visión más completa del desempeño del modelo, se presentan en las Figuras 4.16 y 4.17 la matriz de confusión y las curvas ROC y PR, respectivamente. Estas se basan en las predicciones generadas por el modelo en el conjunto de evaluación correspondiente a DCG-bin. La matriz de confusión (Figura 4.16) exhibe el rendimiento del modelo con una reducida cantidad de errores, tan solo 29, mientras que logra clasificar correctamente un total de 339 muestras. Las curvas (Figura 4.17), a su vez, reflejan y reafirman este desempeño destacado.

Tabla 4.8: Métricas del modelo DT para `DCG-bin`.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	94.06% ± 0.50%	89.47% ± 1.08%	99.17% ± 0.84%	$\overline{434}$
Con riesgo	93.29% ± 0.64%	99.07% ± 0.93%	88.17% ± 1.42%	$\overline{429}$
Accuracy: 93.70% ± 0.56%				
Evaluación				
Sin riesgo	92.30% ± 1.63%	88.09% ± 1.71%	96.98% ± 2.64%	$\overline{186}$
Con riesgo	91.39% ± 1.73%	96.68% ± 2.75%	86.71% ± 2.17%	$\overline{184}$
Accuracy: 91.87% ± 1.67%				

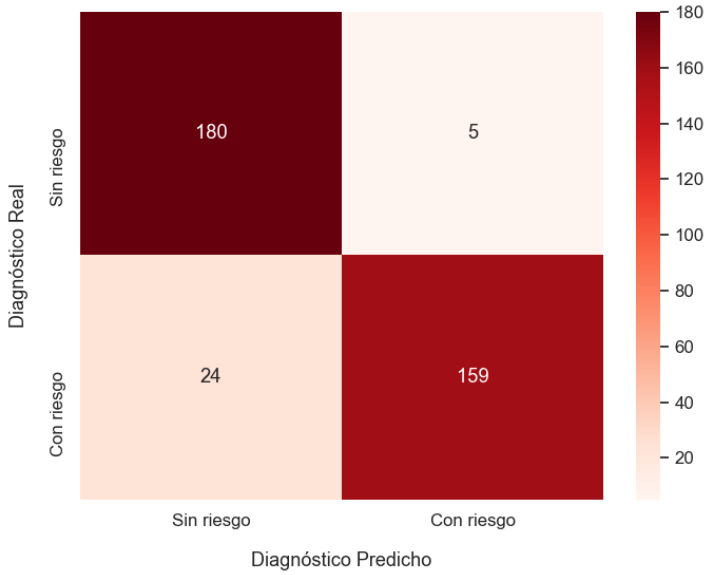
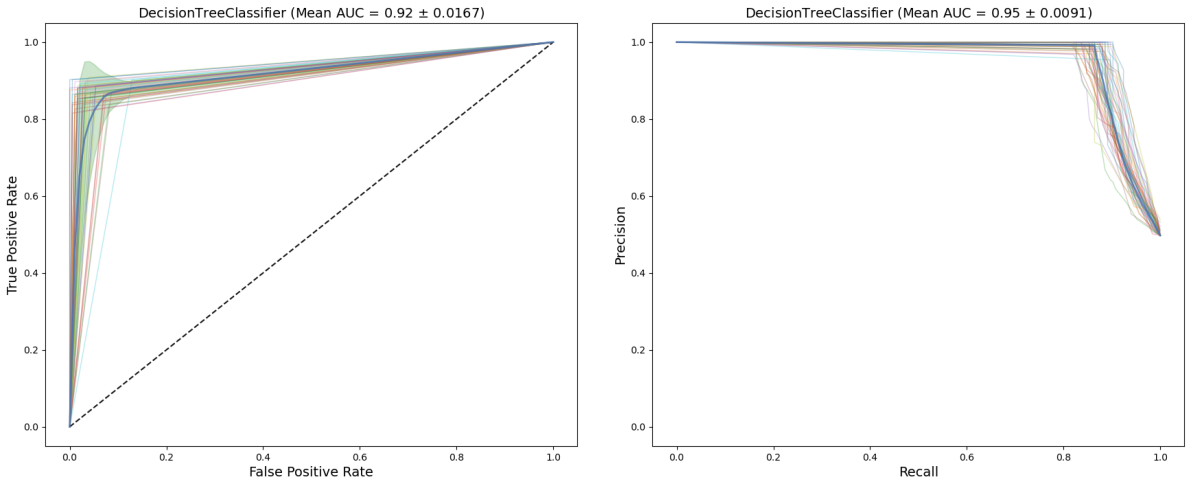


Figura 4.16: Matriz de confusión del modelo DT sobre el conjunto de evaluación para `DCG-bin`.



(a) ROC (b) PR

Figura 4.17: Curvas ROC y PR del modelo DT sobre el conjunto de evaluación para `DCG-bin`.

Luego, se definió un modelo k-NN con 7 vecinos (`n_neighbors = 7`). En la Tabla 4.9 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 79.13% y 69.97% para los conjuntos de entrenamiento y evaluación respectivamente. Este es el peor modelo en términos de *accuracy* al compararlo con los anteriores.

Las Figuras 4.18 y 4.19 exhiben la matriz de confusión y las curvas ROC y PR, generadas a partir del conjunto de evaluación correspondiente a `DCG-bin`, respectivamente. Como era de anticiparse, la matriz de confusión (Figura 4.18) muestra un desempeño inferior en comparación con los modelos presentados anteriormente, evidenciando un mayor número de errores (111). Adicionalmente, las curvas (Figura 4.19) pueden notarse inferiores, en su comportamiento, a las analizadas anteriormente.

Tabla 4.9: Métricas del modelo k-NN para `DCG-bin`.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	79.57% ± 1.01%	78.38% ± 1.23%	80.82% ± 1.64%	$\overline{434}$
Con riesgo	78.67% ± 1.07%	79.99% ± 1.33%	77.42% ± 1.68%	$\overline{429}$
Accuracy: 79.13% ± 1%				
Evaluación				
Sin riesgo	70.66% ± 2.06%	69.46% ± 2.21%	71.99% ± 3.10%	$\overline{186}$
Con riesgo	69.20% ± 2.23%	70.63% ± 2.32%	67.92% ± 3.29%	$\overline{184}$
Accuracy: 69.97% ± 2.01%				

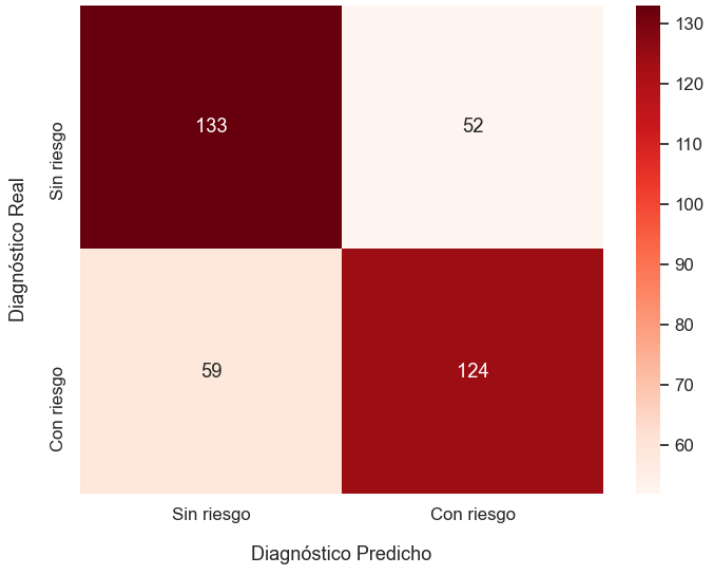
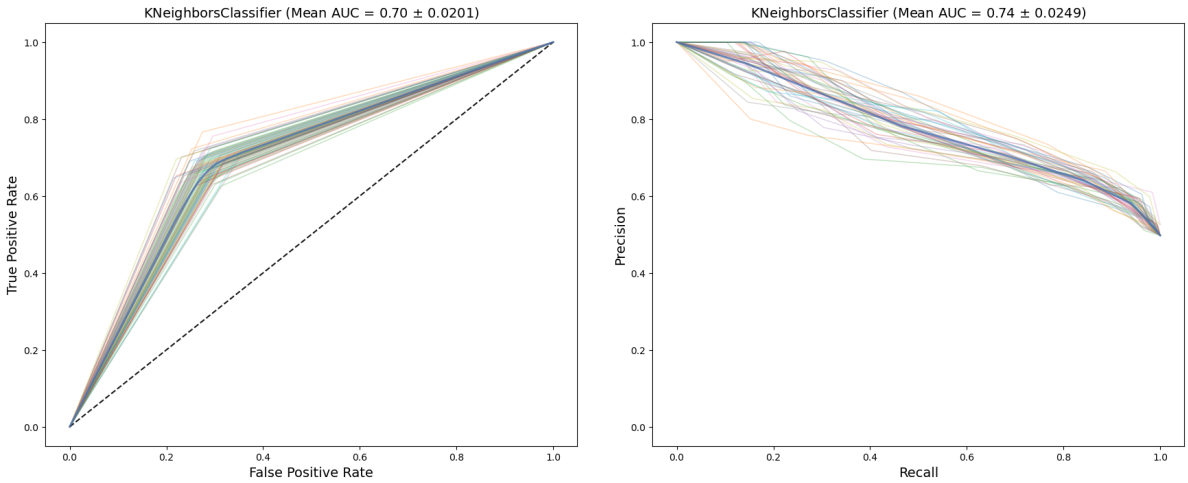


Figura 4.18: Matriz de confusión del modelo k-NN sobre el conjunto de evaluación para `DCG-bin`.



(a) ROC (b) PR

Figura 4.19: Curvas ROC y PR del modelo k-NN sobre el conjunto de evaluación para DCG-bin.

Posteriormente, se entrenó el mismo modelo RF, en término de sus parámetros, que en DCL-bin. En la Tabla 4.10 se resumen los resultados obtenidos. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 93.29% en el conjunto de entrenamiento y del 93.23% en el conjunto de evaluación. Las desviaciones correspondientes a estas métricas son mínimas, similares a las del modelo DT.

Tanto el modelo RF como el DT demuestran un alto grado de efectividad en la clasificación de ambas clases, presentando un valor elevado de *F1-score*, lo que indica un buen equilibrio entre *precision* y *recall*. La Figura 4.21 muestra las curvas ROC y PR generadas a partir del conjunto de evaluación para DCG-bin. Ambas curvas se aproximan al comportamiento ideal, lo que evidencia el excelente rendimiento del modelo RF.

De igual forma, la Figura 4.20 presenta la matriz de confusión, la cual exhibe, hasta el momento, el mejor rendimiento con tan solo 25 errores para la clase “Con riesgo”, logrando clasificar correctamente un total de 344 muestras.

Tabla 4.10: Métricas del modelo RF para DCG-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	93.74% ± 0.42%	88.23% ± 0.75%	100% ± 0%	$\overline{434}$
Con riesgo	92.76% ± 0.55%	100% ± 0%	86.49% ± 0.97%	$\overline{429}$
Accuracy: 93.29% ± 0.48%				

Continúa en la siguiente página

Tabla 4.10 – continuación de la página anterior

	F1-score	precision	recall	N
Evaluación				
Sin riesgo	93.70% ± 0.98%	88.16% ± 1.72%	100% ± 0%	$\overline{186}$
Con riesgo	92.68% ± 1.30%	100% ± 0%	86.38% ± 2.25%	$\overline{184}$
Accuracy: 93.23% ± 1.12%				

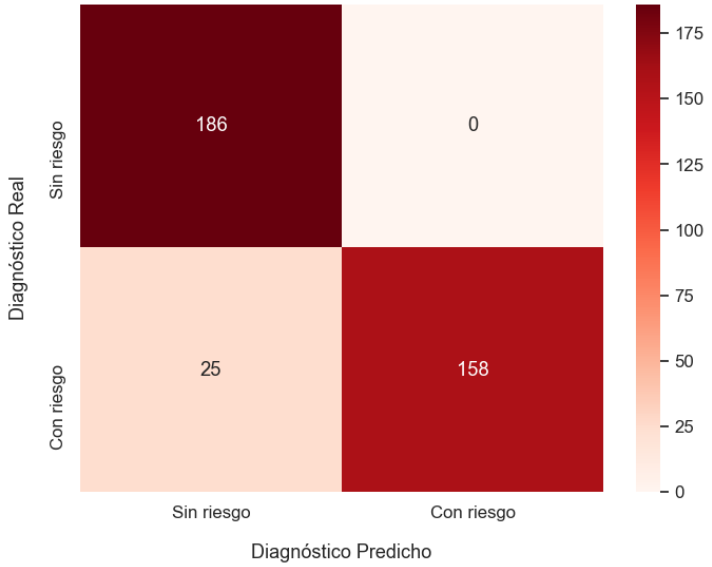
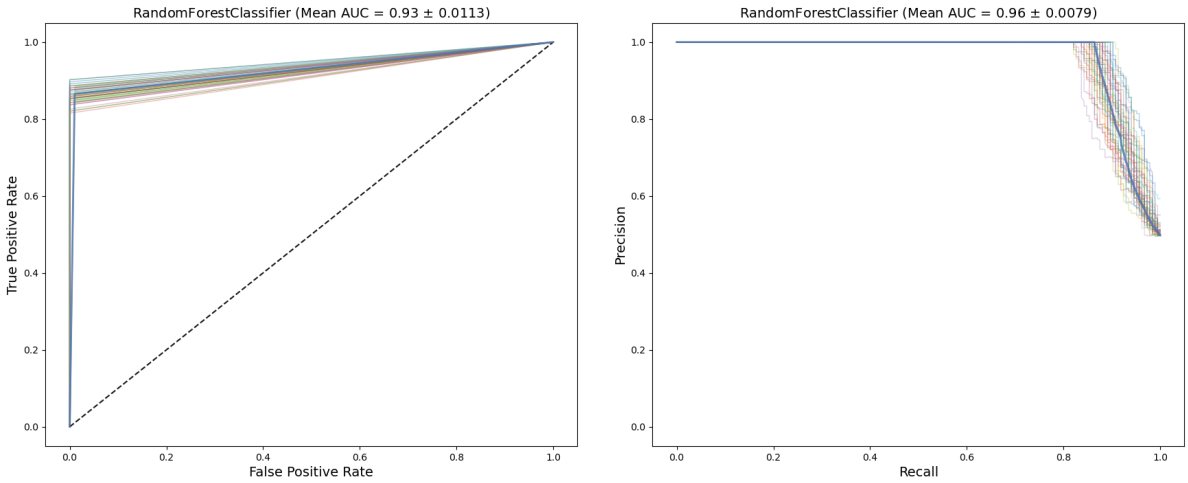


Figura 4.20: Matriz de confusión del modelo RF sobre el conjunto de evaluación para `DCG-bin`.



(a) ROC (b) PR

Figura 4.21: Curvas ROC y PR del modelo RF sobre el conjunto de evaluación para `DCG-bin`.

Con el objetivo de concluir el análisis de los modelos de clasificación para `DCG-bin`, se decidió experimentar el mismo modelo ANN, en términos de su arquitectura y configuración, que en `DCL-bin`.

De acuerdo al modelo propuesto, en la Tabla 4.11 se presentan los resultados obtenidos con un entrenamiento bajo (`epochs = 60`, `batch_size = 16`) y una normalización estándar. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 92.61% en el conjunto de entrenamiento y del 91.05% en el conjunto de evaluación con desviaciones mínimas en ambos casos.

Con el propósito de brindar una visión más completa del desempeño del modelo, se presentan en las Figuras 4.22 y 4.23 la matriz de confusión y las curvas ROC y PR, respectivamente. Estas se basan en las predicciones generadas por el modelo en el conjunto de evaluación correspondiente a `DCG-bin`. La matriz de confusión (Figura 4.22) exhibe un rendimiento similar a la del modelo DT. Las curvas (Figura 4.23), como en los modelos RF y DT, se aproximan al comportamiento ideal, lo que evidencia su muy buen rendimiento.

Tabla 4.11: Métricas del modelo ANN para `DCG-bin`.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	92.94% ± 1.27%	89.40% ± 0.97%	96.78% ± 2.02%	$\overline{434}$
Con riesgo	92.25% ± 1.28%	96.49% ± 2.12%	88.39% ± 1.09%	$\overline{429}$
Accuracy: 92.61% ± 1.27%				

Continúa en la siguiente página

Tabla 4.11 – continuación de la página anterior

	F1-score	precision	recall	N
Evaluación				
Sin riesgo	91.42% ± 1.61%	88.33% ± 1.99%	94.77% ± 2.30%	$\overline{186}$
Con riesgo	90.65% ± 1.76%	94.35% ± 2.34%	87.29% ± 2.45%	$\overline{184}$
Accuracy: 91.05% ± 1.67%				

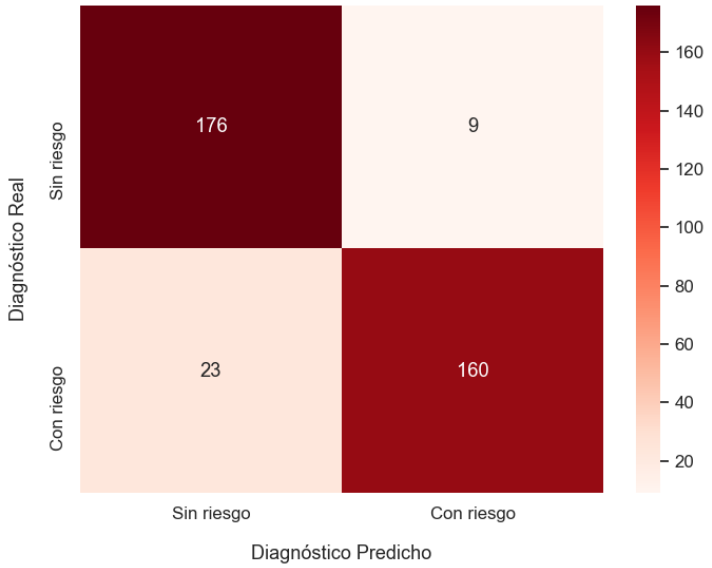
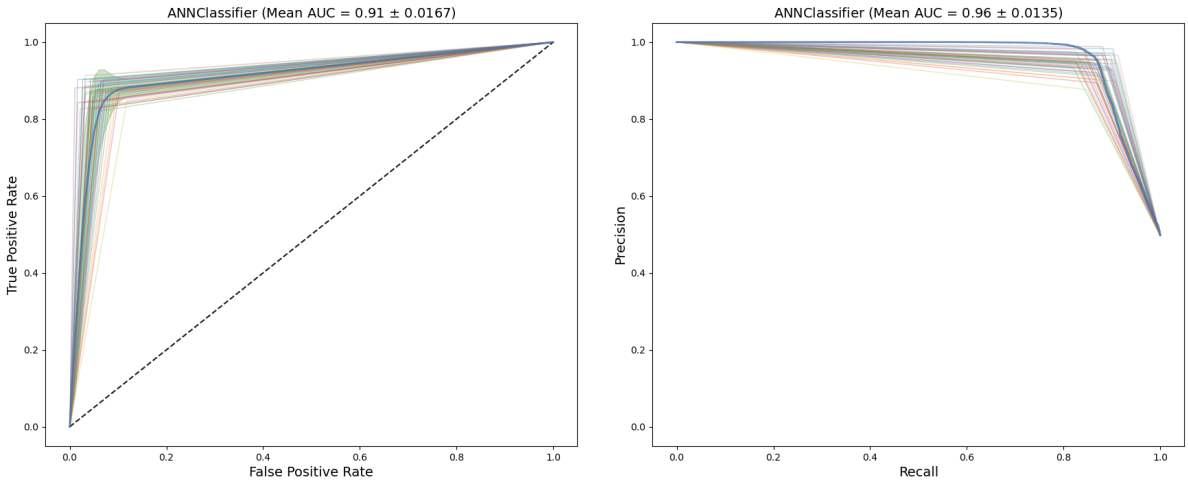


Figura 4.22: Matriz de confusión del modelo ANN sobre el conjunto de evaluación para `DCG-bin`.



(a) ROC (b) PR

Figura 4.23: Curvas ROC y PR del modelo ANN sobre el conjunto de evaluación para DCG-bin.

Finalmente, en la Tabla 4.12, se proporciona un resumen de las métricas obtenidas por los modelos experimentados en el conjunto de evaluación para DCG-bin. A diferencia de casos anteriores en los que se presentaban métricas para ambas clases, esta tabla resume las métricas para la clase "Con riesgo". Los modelos ANN, RF y DT exhiben un grado significativo de efectividad en la clasificación de ambas clases. Estos modelos presentan métricas muy similares entre sí. En contraste, el modelo LOR se ubica por debajo de ellos, seguido por el modelo k-NN, que muestra el menor porcentaje de *accuracy* con un 69.97%. El modelo RF, por otro lado, alcanza el mayor porcentaje de *accuracy* para DCG-bin, con un valor del 93.23%.

Tabla 4.12: Resumen de los modelos experimentados para la clase "Con riesgo" en el conjunto de evaluación sobre DCG-bin.

Clasificador	accuracy	F1-score	precision	recall
RF (max_depth = 2)	93.23% ± 1.12%	92.68% ± 1.30%	100% ± 0%	86.38% ± 2.25%
DT (max_depth = 5, class_weight = ''balanced'')	91.87% ± 1.67%	91.39% ± 1.73%	96.68% ± 2.75%	86.71% ± 2.17%
ANN	91.05% ± 1.67%	90.65% ± 1.76%	94.35% ± 2.34%	87.29% ± 2.45%
LOR (class_weight = ''balanced'')	75.66% ± 4.37%	73.05% ± 5.30%	80.96% ± 4.44%	66.70% ± 6.55%
k-NN (n_neighbors = 7)	69.97% ± 2.01%	69.20% ± 2.23%	70.63% ± 2.32%	67.92% ± 3.29%

4.4 Modelos de clasificación para DC-bin

Bajo el mismo procedimiento utilizado para las segmentaciones anteriores, se llevó a cabo un análisis de DC-bin mediante la utilización de un modelo LOR con balanceo de clases (class_weight=‘‘balanced’’). De manera análoga, se reservó un porcentaje del 30% de DC-bin para la evaluación, empleando la técnica CV descrita en la Sección 4.1 y aplicando una normalización *min-max*. A continuación, la Tabla 4.13 agrupa las métricas obtenidas en ambos conjuntos:

Tabla 4.13: Métricas del modelo LOR para DC-bin.

	F1-score	precision	recall	N
Entrenamiento				
Sin riesgo	65.12% ± 1.57%	57.07% ± 1.05%	76.06% ± 4.79%	$\overline{434}$
Con riesgo	50.34% ± 3.42%	63.67% ± 2.28%	41.99% ± 5.29%	$\overline{429}$
Accuracy: 59.12% ± 1.04%				
Evaluación				
Sin riesgo	63.44% ± 2.82%	55.54% ± 1.74%	74.22% ± 6.04%	$\overline{186}$
Con riesgo	47.94% ± 3.81%	60.88% ± 4.27%	39.92% ± 5.25%	$\overline{184}$
Accuracy: 57.16% ± 2.32%				

La Tabla 4.13 exhibe los resultados del modelo LOR para ambos conjuntos en la media del total de iteraciones (50). En el conjunto de entrenamiento, el modelo obtuvo un *accuracy* del 59.12%, mientras que en el conjunto de evaluación dicha métrica alcanzó un 57.16%. Estos resultados reflejan de manera inmediata el impacto distintivo de emplear exclusivamente los datos clínicos.

La Figura 4.24 muestra la matriz de confusión del modelo LOR para DC-bin sobre el conjunto de evaluación. Se observa que el modelo clasificó correctamente 211 muestras y tuvo 157 errores de clasificación. El modelo posee un número considerable de falsos positivos (47) y falsos negativos (110).

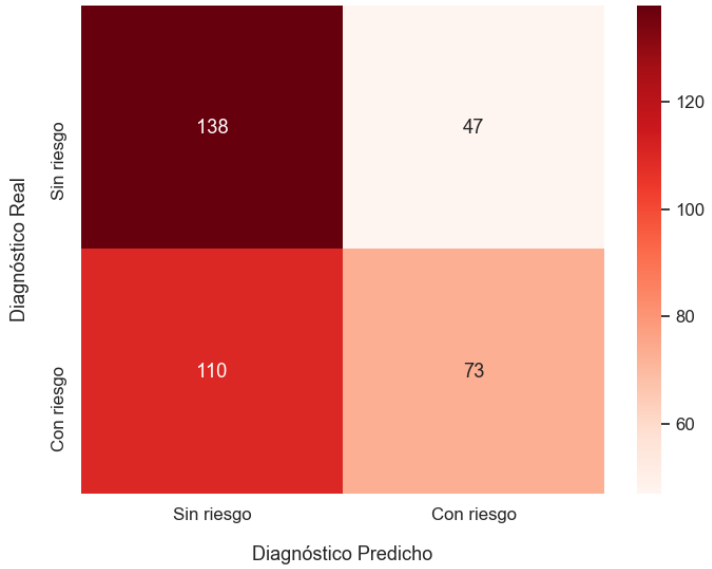


Figura 4.24: Matriz de confusión del modelo LOR sobre el conjunto de evaluación para `DC-bin`.

De acuerdo a los valores de *accuracy* tanto en entrenamiento como evaluación, resulta innecesario profundizar en el análisis de las métricas adicionales.

Aunque el análisis de los pesos asignados por el modelo LOR resulta irrelevante debido al rendimiento demostrado, la Figura 4.25 evidencia una destacada influencia de la variable `le_encontraron_hiperglucemia`. Esto resulta lógico considerando la ausencia de la `glucemia_basal`. La presencia de hiperglucemia actúa como un indicador indirecto hacia un potencial riesgo de diabetes.

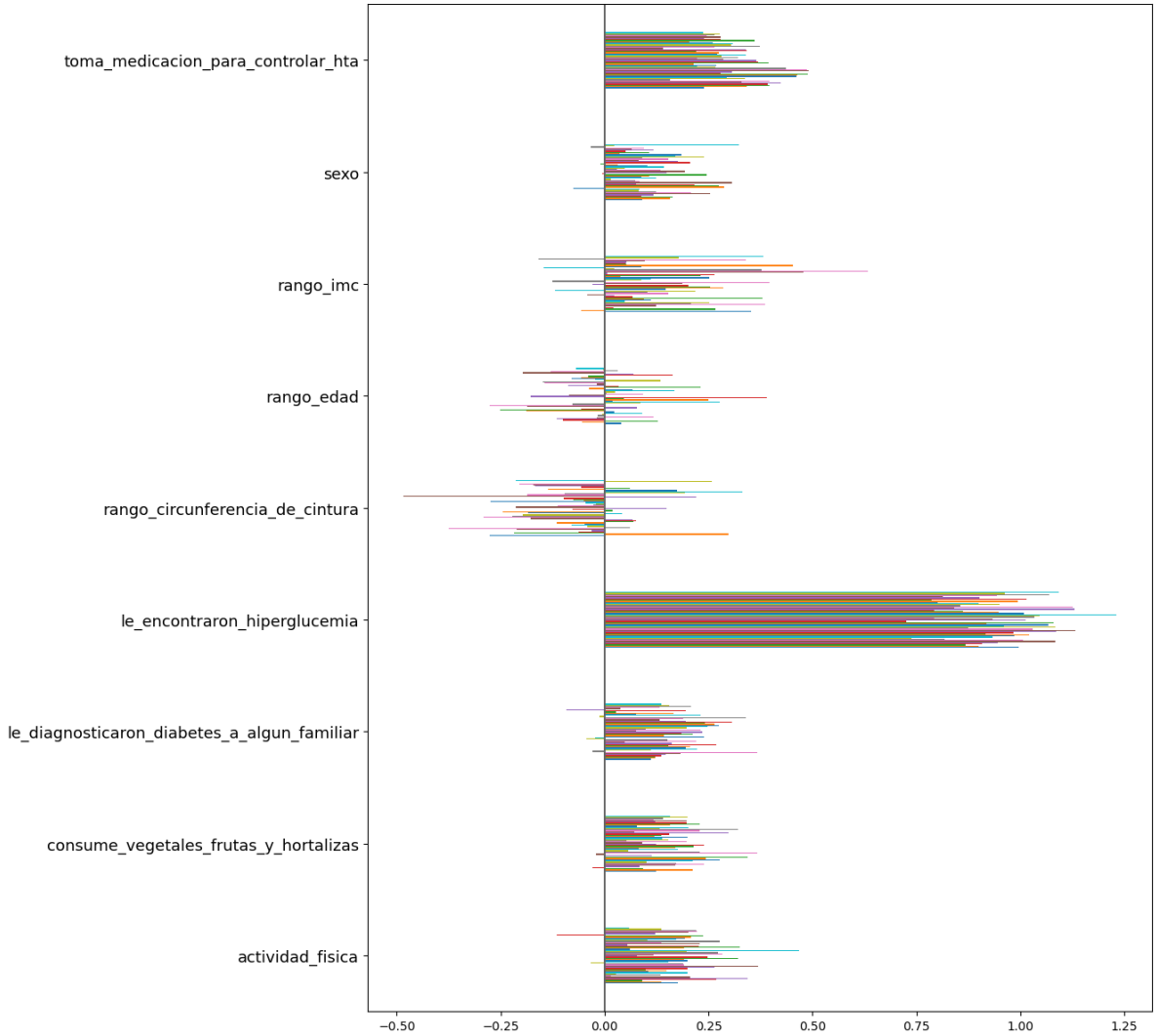
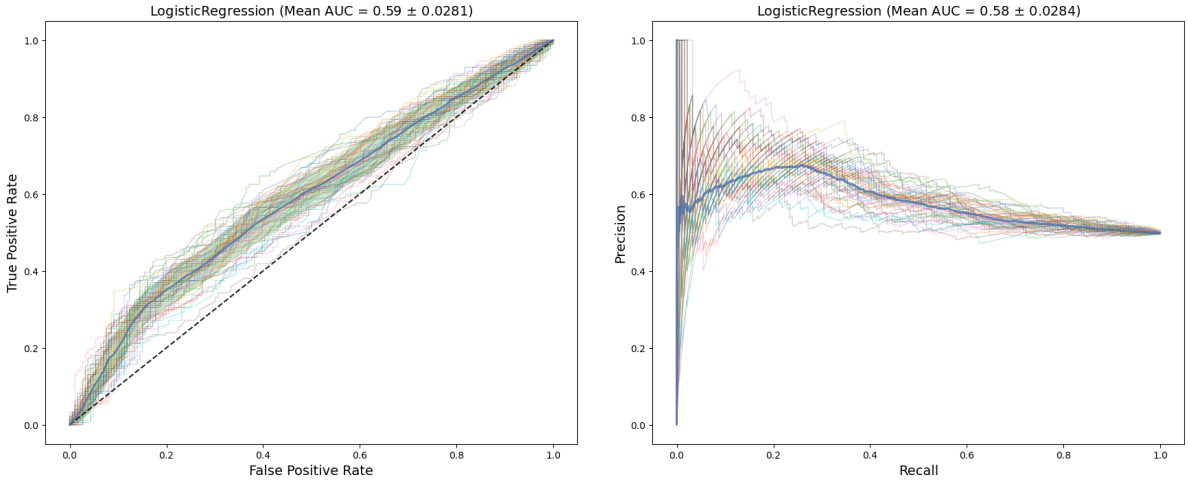


Figura 4.25: Pesos asociados a cada *feature* por el modelo LOR sobre el conjunto de evaluación para `DC-bin`. Cada una de las ejecuciones de CV se encuentra representada por distintas líneas de colores.

A continuación, se presenta en la Figura 4.26 las curvas ROC y PR generadas a partir del conjunto de evaluación para DC-bin. El modelo obtenido posee una media AUC = 0.59 y una media AUC = 0.58 para las curvas ROC y PR respectivamente.

En conjunto, estos hallazgos subrayan las limitaciones del modelo LOR aplicado a la segmentación DC-bin para predecir el riesgo de padecer diabetes.



(a) ROC (b) PR

Figura 4.26: Curvas ROC y PR del modelo LOR sobre el conjunto de evaluación para DC-bin.

La presentación en detalle de las métricas obtenidas mediante la aplicación de diversos modelos a la segmentación DC-bin resulta de escasa relevancia en vista de los resultados alcanzados por el modelo LOR. Sin embargo, en la Tabla 4.14, se proporciona un resumen de las métricas obtenidas por los modelos en el conjunto de evaluación para DC-bin. Esta tabla resume las métricas para la clase “Con riesgo”, reforzando la conclusión de que no es necesario examinar en detalle las demás métricas y curvas de evaluación.

Tabla 4.14: Resumen de los modelos experimentados para la clase "Con riesgo" en el conjunto de evaluación sobre DC-bin.

Clasificador	accuracy	F1-score	precision	recall
ANN	57.55% ± 2.07%	54.27% ± 3.71%	58.61% ± 3.06%	51.05% ± 6.43%
RF (max_depth = 2)	57.33% ± 1.93%	43.61% ± 4.06%	63.77% ± 4.13%	33.42% ± 4.84%
LOR (class_weight = "balanced")	57.16% ± 2.32%	47.94% ± 3.81%	60.88% ± 4.27%	39.92% ± 5.25%
DT (max_depth = 5, class_weight = "balanced")	54.98% ± 2.47%	52.67% ± 5.01%	55.30% ± 3.07%	51.16% ± 9.03%
k-NN (n_neighbors = 7)	54.55% ± 2.45%	53.77% ± 3.03%	54.45% ± 2.53%	53.32% ± 4.79%

4.5 Selección de *features*

El objetivo principal de llevar a cabo un proceso de selección de *features* consiste en identificar y elegir un subconjunto relevante de atributos del conjunto de datos original con el propósito de mejorar el rendimiento de los modelos. Aunque los modelos experimentados exhiben un rendimiento óptimo, se realizó un proceso de selección de *features* manual a partir de la segmentación DCL-bin. El propósito de esta experimentación es determinar los atributos más informativos y útiles para abordar el problema en cuestión, descartando aquellos atributos que resulten irrelevantes, con el objetivo de obtener el mínimo número de variables necesario para lograr un desempeño óptimo.

El proceso involucró la evaluación de modelos previamente experimentados utilizando CV. En cada iteración de este proceso, se eliminó un atributo del conjunto de datos, comenzando desde la segmentación inicial DCL-bin. Este procedimiento se llevó a cabo en varias etapas, comenzando con la eliminación de la variable hemoglobina_glucosilada, seguida de creatinina_basal, colesterol_ldl, triglicéridos, colesterol_hdl y colesterol_total, lo que culminó estructuralmente en la segmentación DCG-bin. Posteriormente, se eliminó la variable glucemia_basal, lo que resultó estructuralmente en la segmentación final DC-bin. El término de igualdad estructural refiere a que las segmentaciones comparten los mismos *features*, aunque no poseen la misma cantidad de datos. Estas selecciones sólo contienen 503 registros de los 1233 que poseen DCG-bin y DC-bin.

Los resultados de este proceso se presentan en la Tabla 4.15. En la primera columna, se incluye un identificador para cada conjunto de datos. La columna Dataset describe el conjunto de datos utilizado, que se obtiene a partir del conjunto anterior con la eliminación progresiva de los atributos mencionados. Las columnas N e in indican la cantidad de muestras y los atributos de entrada correspondientes a cada modelo, respectivamente.

Asimismo, la Tabla 4.15, presenta la agrupación de los *accuracy* obtenidos sobre

el conjunto de evaluación para cada uno de los modelos. Como es posible observar, el modelo LOR mantiene un *accuracy* que oscila entre el 85% y el 87% hasta la selección D_6 inclusive. Por otro lado, k-NN incrementa su *accuracy*, pasando del 71% al 79% utilizando la selección D_6 inclusive. En tanto, RF, ANN, y DT, se mantienen relativamente estables al reducir los atributos de laboratorio. A pesar de que los incrementos en el rendimiento de los modelos no son significativos, todos ellos muestran comportamientos similares durante el proceso de eliminación de los atributos de laboratorio.

Tabla 4.15: *Accuracy* de los modelos experimentados sobre el conjunto de evaluación utilizando CV.

	Dataset (D)	N	in	LOR	DT	k-NN	RF	ANN
D_0	DCL-bin	503	16	85.56%	93.43%	71.62%	94.64%	91.14%
D_1	D_0 - hem___glucosilada	503	15	86.64%	93.27%	70.01%	94.75%	91.60%
D_2	D_1 - creatinina_basal	503	14	86.58%	93.48%	70.61%	94.87%	91.66%
D_3	D_2 - colesterol_ldl	503	13	86.77%	93.34%	71.15%	94.91%	91.52%
D_4	D_3 - trigliceridos	503	12	87.15%	93.01%	74.05%	94.86%	91.68%
D_5	D_4 - colesterol_hdl	503	11	87.06%	92.82%	75.63%	94.95%	92.08%
D_6	D_5 - colesterol_total	503	10	87.63%	93.72%	79.70%	94.85%	92.70%
D_7	D_6 - glucemia_basal	503	9	59.21%	57.21%	56.50%	59.68%	58.61%

Es posible reafirmar la importancia de la variable `glucemia_basal` en la predicción del riesgo de diabetes. Esto se evidencia claramente en la última fila de la Tabla 4.15, donde se aprecia una disminución abrupta en el rendimiento de los modelos al eliminar dicha variable. Este hallazgo subraya la relevancia crítica de la `glucemia_basal` en la evaluación y detección del riesgo de diabetes, ya que su exclusión conlleva un impacto significativo en el rendimiento de los modelos predictivos.

4.6 Modelos de regresión para DCL

Con el fin de diversificar el conjunto de experimentos, se tomó la decisión de abordar el problema original de detección de riesgo de diabetes desde una perspectiva diferente, utilizando el enfoque de regresión. A pesar de que este problema podría considerarse como un caso de clasificación multiclase, se optó por tratarlo como un problema de regresión, en el que la variable de salida $f(x)$ se mapea en tres posibles valores:

$$f(x) = \begin{cases} Normal & \text{si } x = 0 \\ PDM & \text{si } x = 1 \\ DM & \text{si } x = 2 \end{cases}$$

Asimismo, entre las métricas definidas para evaluar los modelos de regresión experimentados, se incluye el cálculo del *accuracy*. Esta métrica se obtiene mediante la función de decisión $D(x)$, la cual se define de la siguiente manera:

$$D(x) = \begin{cases} 0 & \text{si } x < 0.5 \\ 1 & \text{si } 0.5 \leq x \leq 1.5 \\ 2 & \text{si } x > 1.5 \end{cases}$$

Para llevar a cabo estas experimentaciones, se optó por emplear la segmentación `DCL`, que contiene la misma cantidad de registros que `DCL-bin`. En esta etapa inicial, se procedió a realizar un análisis de `DCL` mediante la implementación de un modelo LR. Dicho modelo posee múltiples variables de entrada y un valor numérico como predicción. Se reservó un porcentaje del 30% de `DCL` para la evaluación, empleando la técnica CV descrita en la Sección 4.1 y aplicando una normalización *min-max*. A continuación, la Tabla 4.16 agrupa las métricas obtenidas en ambos conjuntos:

Tabla 4.16: Métricas del modelo LR para `DCL`.

	MSE	RMSE	MAE	MAE por clase	R2
Entrenamiento					
	0.13 ± 0.01	0.36 ± 0.01	0.28 ± 0.01	Normal ⇒ 0.29	0.61 ± 0.03
				PDM ⇒ 0.24	
				DM ⇒ 0.61	
Accuracy: 87.67% ± 1.25%					
Evaluación					
	0.15 ± 0.02	0.38 ± 0.03	0.30 ± 0.01	Normal ⇒ 0.31	0.56 ± 0.07
				PDM ⇒ 0.26	
				DM ⇒ 0.66	
Accuracy: 86.42% ± 2.65%					

La Tabla 4.16 presenta los resultados del modelo LR para ambos conjuntos en la media del total de iteraciones (50). En el conjunto de entrenamiento, el modelo LR obtuvo un *accuracy* del 87.67%, con desviaciones inferiores al 1.5%. Por otro lado, en el conjunto de evaluación, dicha métrica alcanzó un 86.42%, con desviaciones menores al 2.7%. El

modelo tiene una capacidad aceptable para generalizar a nuevos datos, clasificando correctamente aproximadamente 8 de cada 10 muestras.

En relación a las métricas MSE, RMSE y MAE, se considera que un valor menor indica un mejor ajuste del modelo. Al analizan dichas métricas de error para el modelo LR, se puede concluir que el mismo muestra un buen rendimiento.

Por otra parte, un valor de métrica R^2 cercano a 1 indica un mejor ajuste del modelo. En este caso, los valores obtenidos fueron de 0.61 y 0.56 para los conjuntos de entrenamiento y evaluación, respectivamente.

Dado que el conjunto de datos presenta un desequilibrio, es posible que el modelo tenga dificultades para predecir correctamente las clases minoritarias debido a la falta de muestras de entrenamiento. Esto puede afectar el rendimiento del modelo, lo cual se refleja en el MAE por clase. Se observa que el MAE para la clase "DM" en el conjunto de evaluación es significativamente mayor en comparación con las otras clases, lo cual indica que el modelo tiene dificultades para predecir correctamente dicha clase.

Para finalizar el análisis del modelo LR aplicado a `DCL`, se presenta en la Figura 4.27 la distribución de pesos asignados a cada atributo. Como era de esperar, se observa una clara predominancia de `glucemia_basal`, seguida por `colesterol_total` y `hemoglobina_glucosilada`.

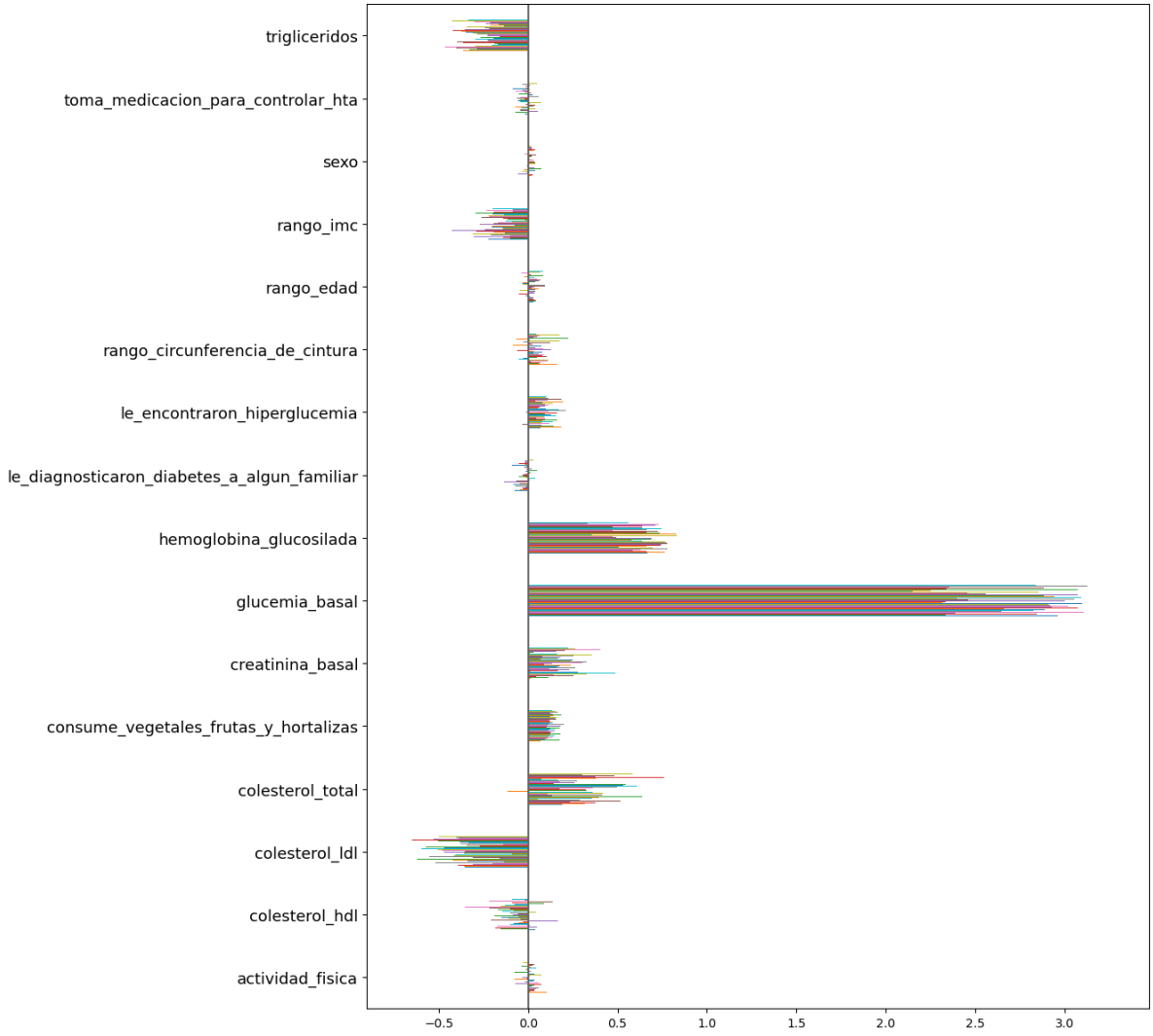


Figura 4.27: Pesos asociados a cada *feature* por el modelo LR sobre el conjunto de evaluación para DCL. Cada una de las ejecuciones de CV se encuentra representada por distintas líneas de colores.

A fines comparativos, se exponen a continuación las métricas correspondientes a los modelos restantes, los cuales fueron entrenados y evaluados de la misma forma que el modelo LR.

De esta forma, se definió un modelo de regresión DT con una profundidad máxima de 5 (`max_depth = 5`). En la Tabla 4.17 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 97.23% y 91.14% para los conjuntos de entrenamiento y evaluación respectivamente. Es relevante notar que las desviaciones observadas en estas métricas son mínimas, indicando una consistencia en el rendimiento del modelo. Este resultado representa una mejora significativa con respecto al modelo LR anteriormente presentado.

Tabla 4.17: Métricas del modelo de regresión DT para DCL.

	MSE	RMSE	MAE	MAE por clase	R2
Entrenamiento					
	0.03 ± 0.01	0.16 ± 0.03	0.05 ± 0.01	Normal ⇒ 0.05	0.92 ± 0.02
				PDM ⇒ 0.04	
				DM ⇒ 0.10	
Accuracy: 97.23% ± 0.74%					
Evaluación					
	0.11 ± 0.03	0.32 ± 0.05	0.11 ± 0.02	Normal ⇒ 0.10	0.68 ± 0.09
				PDM ⇒ 0.10	
				DM ⇒ 0.47	
Accuracy: 91.14% ± 2%					

Luego, se definió un modelo de regresión k-NN con 7 vecinos (`n_neighbors = 7`). En la Tabla 4.18 se presentan los resultados obtenidos, en la media del total de iteraciones, con un *accuracy* del 76.59% y 67.85% para los conjuntos de entrenamiento y evaluación respectivamente. Como el posible observar, hasta el momento, este es el peor modelo en términos de *accuracy* al compararlo con los anteriores experimentados.

Tabla 4.18: Métricas del modelo de regresión k-NN para DCL.

	MSE	RMSE	MAE	MAE por clase	R2
Entrenamiento					
	0.18 ± 0.01	0.42 ± 0.01	0.34 ± 0.02	Normal ⇒ 0.31 PDM ⇒ 0.31 DM ⇒ 0.95	0.46 ± 0.03
Accuracy: 76.59% ± 1.80%					
Evaluación					
	0.24 ± 0.02	0.49 ± 0.02	0.40 ± 0.02	Normal ⇒ 0.37 PDM ⇒ 0.36 DM ⇒ 1.14	0.27 ± 0.06
Accuracy: 67.85% ± 3.41%					

De igual forma, se entrenó el modelo de regresión RF, configurado con el parámetro `max_depth = 2`. En la Tabla 4.19 se resumen los resultados obtenidos. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 94.52% en el conjunto de entrenamiento y del 93.71% en el conjunto de evaluación. Las desviaciones correspondientes a estas métricas son mínimas.

Tabla 4.19: Métricas del modelo de regresión RF para DCL.

	MSE	RMSE	MAE	MAE por clase	R2
Entrenamiento					
	0.06 ± 0.01	0.23 ± 0.02	0.10 ± 0.01	Normal ⇒ 0.09 PDM ⇒ 0.09 DM ⇒ 0.43	0.83 ± 0.03
Accuracy: 94.52% ± 0.63%					
Evaluación					
	0.07 ± 0.02	0.27 ± 0.04	0.12 ± 0.01	Normal ⇒ 0.10 PDM ⇒ 0.10 DM ⇒ 0.51	0.78 ± 0.07
Accuracy: 93.71% ± 1.60%					

De acuerdo con las experimentaciones llevadas a cabo en los modelos de clasificación y con el objetivo de concluir el análisis de los modelos de regresión para DCL, se optó por

realizar una experimentación con un modelo ANN. De esta forma, se construyó una ANN con una sola capa oculta. En dicha capa, se especificó un tamaño de 100 neuronas y se utilizó la función de activación `ReLU`. Asimismo, se aplicó una regularización L2 de 0.1 permitiendo controlar el sobreajuste del modelo. La última capa, consta de una única neurona encargada de generar la salida del modelo de regresión.

Posteriormente, se procedió a compilar el modelo, utilizando un optimizador `Adamax` con una tasa de aprendizaje de 0.001. Como función de pérdida se utilizó el MSE, adecuado para abordar problemas de regresión.

Conforme al modelo propuesto, los resultados correspondientes se exponen en la Tabla 4.20. Estos resultados se derivan de un entrenamiento bajo (`epochs = 60`, `batch_size = 16`) y una normalización estándar. En la media del total de iteraciones, el modelo alcanzó un porcentaje de *accuracy* del 98.01% en el conjunto de entrenamiento y del 88.79% en el conjunto de evaluación con desviaciones mínimas en ambos casos. No obstante, puede observarse una disminución significativa tanto en el *accuracy* como en R^2 entre el conjunto de entrenamiento y evaluación, lo cual sugiere la posibilidad de un sobreajuste en el modelo ANN. Cabe señalar, sin embargo, que el propósito de la presente investigación no reside en la búsqueda del modelo óptimo, sino en la exploración de las diversas segmentaciones propuestas bajo algunos modelos comúnmente utilizados.

Tabla 4.20: Métricas del modelo ANN para DCL.

	MSE	RMSE	MAE	MAE por clase	R2
Entrenamiento					
	0.03 ± 0.02	0.18 ± 0.05	0.13 ± 0.04	Normal ⇒ 0.12	0.89 ± 0.06
				PDM ⇒ 0.13	
				DM ⇒ 0.24	
Accuracy: 98.01% ± 1.81%					
Evaluación					
	0.11 ± 0.02	0.33 ± 0.03	0.24 ± 0.02	Normal ⇒ 0.20	0.68 ± 0.06
				PDM ⇒ 0.24	
				DM ⇒ 0.54	
Accuracy: 88.79% ± 2.48%					

Finalmente, en la Tabla 4.21, se proporciona un resumen de las métricas obtenidas por los modelos experimentados en el conjunto de evaluación para DCL. Los modelos ANN, RF y DT exhiben un grado significativo de efectividad en la regresión de valores para DCL. Estos modelos presentan métricas muy similares entre sí. En contraste, el modelo LR se ubica por debajo de ellos, seguido por el modelo k-NN, que muestra el menor porcentaje de *accuracy* con un 67.85%. El modelo RF, por otro lado, alcanza el mayor porcentaje de *accuracy* para DCL, con un valor del 93.71%.

El porcentaje de *accuracy* obtenido para los modelos ANN, RF y DT indica su gran capacidad para generalizar hacia nuevos datos. Asimismo, en términos de MAE por clase, todos los modelos experimentados muestran mejores medidas para las clases "Normal" y "PDM" que para "DM". Esto sugiere la dificultad de identificar pacientes con "DM" debido a la escasez de muestras disponibles para esta condición. Las restantes métricas presentan valores alineados con el rendimiento obtenido por los modelos.

Tabla 4.21: Resumen de los modelos experimentados en el conjunto de evaluación sobre DCL.

Regresor	accuracy	MSE	RMSE	MAE	R2
RF (max_depth = 2)	93.71% ± 1.60%	0.07 ± 0.02	0.27 ± 0.04	0.12 ± 0.01	0.78 ± 0.07
DT (max_depth = 5)	91.14% ± 2%	0.11 ± 0.03	0.32 ± 0.05	0.11 ± 0.02	0.68 ± 0.09
ANN	88.79% ± 2.48%	0.11 ± 0.02	0.33 ± 0.03	0.24 ± 0.02	0.68 ± 0.06
LR	86.42% ± 2.65%	0.15 ± 0.02	0.38 ± 0.03	0.30 ± 0.01	0.56 ± 0.07
k-NN (n_neighbors = 7)	67.85% ± 3.41%	0.24 ± 0.02	0.49 ± 0.02	0.40 ± 0.02	0.27 ± 0.06

4.7 Análisis comparativo

Con el fin de proporcionar una conclusión y una comparación definitiva de cada segmentación evaluada en el presente capítulo, la Tabla 4.22 resume los modelos experimentados para cada una de ellas sobre el conjunto de evaluación. En ella se presentan cada una de las segmentaciones propuestas, los modelos de clasificación o regresión experimentados, y el *accuracy* obtenido por cada uno. Debido a que el análisis del *accuracy* podría no ser suficiente, se agregaron otras métricas particulares para problemas de clasificación y regresión.

A partir del análisis de la información proporcionada en la Tabla 4.22, es posible notar que no hay diferencias significativas entre los mejores valores de *accuracy* y *F1-score* conseguidos por los clasificadores para las segmentaciones DCL-bin y DCG-bin. Aunque no es (del todo) apropiado comparar resultados de modelos entrenados con conjuntos de datos diferentes, esta cuestión podría tener incidencia en el costo de llevar a la práctica los modelos, considerando que la obtención de variables de laboratorio no es un proceso gratuito ni sencillo. Para poder dilucidarla, sería necesario disponer de un mayor número de registros sin valores nulos.

Adicionalmente, se ha podido constatar en la Sección 4.5 que el atributo glucemia_basal resulta determinante para el desempeño de los modelos. La omisión de este atributo en el conjunto de datos tiene un impacto crítico en su rendimiento. Este fenómeno es posible observarlo en la Tabla 4.22, donde se observa una marcada diferencia en el rendimiento de los modelos de clasificación cuando se utiliza únicamente la información clínica a través de la segmentación DC-bin.

Tabla 4.22: Resultados comparativos de los modelos para las segmentaciones propuestas sobre el conjunto de evaluación.

Dataset	Clasificador / Regresor	Modelo	Accuracy	Otras métricas
DCL-bin	Clasificador	RF	94.56% ± 1.60%	F1-score: 94.77% ± 1.60%
		DT	93.46% ± 1.58%	F1-score: 93.84% ± 1.50%
		ANN	91.22% ± 2.06%	F1-score: 91.87% ± 1.90%
		LOR	85.56% ± 3.21%	F1-score: 86.01% ± 3.30%
		k-NN	71.62% ± 3.82%	F1-score: 72.97% ± 4%
DCG-bin	Clasificador	RF	93.23% ± 1.12%	F1-score: 92.68% ± 1.30%
		DT	91.87% ± 1.67%	F1-score: 91.39% ± 1.73%
		ANN	91.05% ± 1.67%	F1-score: 90.65% ± 1.76%
		LOR	75.66% ± 4.37%	F1-score: 73.05% ± 5.30%
		k-NN	69.97% ± 2.01%	F1-score: 69.20% ± 2.23%
DC-bin	Clasificador	ANN	57.55% ± 2.07%	F1-score: 54.27% ± 3.71%
		RF	57.33% ± 1.93%	F1-score: 43.61% ± 4.06%
		LOR	57.16% ± 2.32%	F1-score: 47.94% ± 3.81%
		DT	54.98% ± 2.47%	F1-score: 52.67% ± 5.01%
		k-NN	54.55% ± 2.45%	F1-score: 53.77% ± 3.03%
DCL	Regresor	RF	93.71% ± 1.60%	MAE: 0.12 ± 0.01 MSE: 0.07 ± 0.02
		DT	91.14% ± 2%	MAE: 0.11 ± 0.02 MSE: 0.11 ± 0.03
		ANN	88.79% ± 2.48%	MAE: 0.24 ± 0.02 MSE: 0.11 ± 0.02
		LR	86.42% ± 2.65%	MAE: 0.30 ± 0.01 MSE: 0.15 ± 0.02
		k-NN	67.85% ± 3.41%	MAE: 0.40 ± 0.02 MSE: 0.24 ± 0.02

Desde un punto de vista abstracto y con el objetivo de identificar un cierto patrón en el rendimiento general de los modelos, algunos de ellos obtuvieron muy buenos rendimientos tanto en clasificación como regresión. En particular, RF, DT y ANN han demostrado un alto poder predictivo, con valores considerables en las métricas relevantes para cada uno de los problemas abordados. En ese sentido, resulta importante aclarar que los modelos

propuestos no pretenden reemplazar a las PTOG como mecanismo de diagnóstico de DT2 y PDM. Al ser enfermedades de difícil detección precoz, estos modelos buscan identificar aquellas personas de la población Argentina que tengan alta probabilidad de tenerlas y desconozcan su condición. Para confirmar el diagnóstico, las personas identificadas deberán realizar eventualmente una PTOG. Los modelos ayudarían a identificar a quienes deben realizarlo y suplirían la ausencia de herramientas de este tipo.

Por su parte, el modelo LOR continúa a los tres modelos con mejores rendimientos tanto en clasificación como regresión. El mismo modelo aplicado a `DCG-bin` presenta un desempeño inferior del 10% respecto al rendimiento en `DCL-bin`. Sin embargo, se tiene el mismo patrón de errores en las matrices de confusión para ambas segmentaciones donde, el modelo, presentó más falsos negativos que falsos positivos. Sin importar la segmentación empleada, el modelo k-NN muestra el rendimiento más deficiente en comparación con los demás.

A fin de llevar a cabo un análisis y una comparación a nivel segmentación, `DCL-bin` se caracteriza por incluir tanto la información clínica como la información de laboratorio de cada paciente. No obstante, es importante tener en consideración que la viabilidad de esta segmentación depende de la disponibilidad de información de laboratorio actualizada. Además, la obtención de dicha información conlleva una inversión considerable en términos de tiempo y costos para cada paciente. A pesar de que se hayan logrado modelos con un buen rendimiento utilizando esta segmentación, es posible que las personas no deseen compartir su información de laboratorio debido a su naturaleza sensible.

Por lo tanto, considerando la importancia de la variable `glucemia_basal`, la segmentación `DCG-bin` se centra exclusivamente en dicho atributo. Una de las ventajas significativas de este enfoque es que, a diferencia de `DCL-bin`, se preserva el número total de registros (1233). Además de esta ventaja, se observa que `DCG-bin` presenta un desequilibrio de clases menor en comparación con `DCL-bin`.

Con el objetivo de evaluar la viabilidad de desarrollar modelos capaces de predecir el riesgo de diabetes sin depender de información de laboratorio costosa, se tomó la decisión de utilizar únicamente la información clínica del conjunto de datos, lo que condujo a la generación de la segmentación `DC`. Una de las ventajas evidentes de este enfoque reside en el uso exclusivo de atributos clínicos, cuyos valores pueden ser proporcionados fácilmente por los pacientes. Sin embargo, esta segmentación excluye el atributo más influyente, como es el caso de la `glucemia_basal`, lo cual derivó en una baja significativa en el rendimiento de los modelos experimentados.

La elección de la segmentación `DCL` ofrece la ventaja de permitir una mayor precisión en la detección no solo del riesgo de diabetes, sino también de su etapa específica. No obstante, este enfoque presenta un desequilibrio notable en el conjunto de datos. En la Tabla 3.8, es posible observar el número reducido de muestras correspondientes a “DM”. Esta precondición resultó en modelos con una mayor dificultad para predecir pacientes con esta condición.

Finalmente, es importante destacar la decisión de agrupar “PDM” y “DM” como una única clase, lo cual favoreció al balanceo y simplificó el problema al volverlo de clasificación binaria. Sin embargo, esta simplificación conlleva costo de no poder distinguir entre los casos de PDM y DM. Desde una perspectiva médica, esta distinción

puede no ser tan grave, ya que lo fundamental es identificar a las personas en riesgo, independientemente de la etapa específica de la enfermedad.

CAPÍTULO 5

CONCLUSIONES Y TRABAJOS FUTUROS

En el presente capítulo, se exponen las conclusiones principales que emergen a partir de las experimentaciones realizadas (Sección 5.1). Luego, se presentan posibles líneas de investigación en pos de ampliar los temas abordados en la presente investigación (Sección 5.2).

5.1 Conclusiones

Tanto la Diabetes Mellitus (DM) como la Prediabetes (PDM) representan una creciente amenaza global para la salud, incluyendo a la población Argentina. En este contexto, resulta esencial enfatizar la relevancia de la detección temprana de estas estas condiciones, dado que en numerosas ocasiones los síntomas tienden a manifestarse de manera "silenciosa", es decir, de forma asintomática. Asimismo, existe una falta de conocimiento de los factores de riesgo asociados lo que hace aún más difícil su detección. La gran cantidad de personas que padecen esta enfermedad sin conocimiento alguno hasta alcanzar niveles irreversibles es un reflejo de la complejidad para su detección temprana. La DM y PDM no diagnosticada a tiempo o mal controlada puede conducir al desarrollo y progresión de complicaciones graves y costos de salud significativos. Sin embargo, las intervenciones tempranas, como cambios en el estilo de vida, pueden retrasar o prevenir la enfermedad. Por esta razón, la detección temprana para su control resulta ser un desafío importante.

En este contexto, el Aprendizaje Automático (ML) se ha empleado con éxito para mejorar la identificación temprana de diversas enfermedades. Esta contribución se traduce en un respaldo valioso para los profesionales de la salud al facilitar la adopción de decisiones más fundamentadas y eficaces, lo que repercute de manera positiva en la mejora del diagnóstico y consecuentemente en la calidad general de la atención médica. En Argentina, el Programa Piloto de Prevención Primaria de Diabetes en la provincia de Buenos Aires (PPDBA), cuenta con una base de datos que podría ser utilizada para la investigación, el desarrollo y la comparación de diversos modelos predictivos orientados a la identificación de personas con un elevado riesgo de padecer DM y PDM en su población. Es por lo que en esta tesina se propuso como objetivo desarrollar y evaluar modelos predictivos que permitan una detección efectiva de aquellas personas con DM y PDM para la población Argentina, haciendo uso del conjunto de datos proporcionado por el programa PPDBA.

A lo largo de esta investigación, se proporcionó el marco teórico referido a los dos temas principales: la DM y el ML, además de presentar un análisis del estado del arte actual. Este enfoque se llevó a cabo con el propósito de explorar y establecer el marco global existente en el que se llevaron a cabo las experimentaciones subsiguientes.

A continuación, para lograr un entendimiento y caracterización del conjunto de datos disponible, se realizó un análisis descriptivo inicial del mismo considerando toda su información. Luego, fue necesario realizar un cuidadoso preprocesamiento del conjunto de datos, el cual abarcó la eliminación de ciertos atributos, el análisis de valores atípicos y nulos y la creación de una nueva variable de clase que permitió binarizar el problema en cuestión. Esto significó dividir el conjunto de datos en personas sin riesgo de tener PDM o DM y las que sí lo tienen. Asimismo, en lugar de plantear un único escenario que abarcara todo el conjunto de datos disponible, se decidió generar distintas segmentaciones o subdivisiones del mismo (`DCL/DCL-bin`, `DCG/DCG-bin`, `DC/DC-bin` y `DL/DL-bin`) considerando el compromiso entre cantidad de variables y de registros disponibles, además del propósito del modelo. El objetivo de este particionamiento consistió en generar y evaluar modelos que presenten diferentes relaciones de costo-beneficio.

No obstante, desde una perspectiva médica, se llegó a la conclusión de que las

segmentaciones que solo consideran la información de laboratorio (DL/DL-bin) carecían de sentido. Ante esto, se descartaron dichas segmentaciones en los experimentos abordados.

Posteriormente, se procedió con la experimentación de los modelos, centrándose en las siguientes segmentaciones, las cuales se dividen en dos grandes problemas: el problema de clasificación binaria (DCL-bin, DCG-bin y DC-bin) y el problema de regresión (DCL). La perspectiva del problema de clasificación consiste en identificar pacientes con o sin riesgo de diabetes; mientras que el enfoque de regresión se desprende del problema original de detección de riesgo de diabetes en su etapa específica ("Normal", "PDM" y "DM").

Tras la definición de las segmentaciones y la contextualización de los problemas a ser abordados, se procedió a realizar una serie de experimentaciones empleando diversos modelos predictivos en cada una de ellas. Particularmente, se desarrollaron modelos de Redes Neuronales Artificiales (ANN), *Random Forest* (RF), Árbol de Decisión (DT), Regresión Lineal (LR), Regresión Logística (LOR) y K vecinos más cercanos (k-NN). Asimismo, se realizó un proceso de selección de *features* manual a partir de la segmentación DCL-bin con el objetivo de determinar los atributos más informativos y útiles para abordar el problema en cuestión. Una vez analizados los resultados derivados de las experimentaciones previamente mencionadas, fue posible extraer las siguientes conclusiones:

- Partiendo del modelo LOR, el cual permite visualizar los pesos asignados a cada atributo, y de las segmentaciones DCL-bin y DCG-bin, se pudo observar que la mayoría de los *features* carecen de relevancia para el problema de clasificación. No obstante, se identificó una lógica discernible en la importancia relativa de dos características específicas: la glucemia_basal sobresalió como el atributo más influyente en la predicción del riesgo de diabetes, seguida por la hemoglobina_glucosilada, que mostró un impacto de menor predominancia. En la ausencia de información de laboratorio, mediante la segmentación DC-bin, se evidenció una destacada influencia de la variable le_encontraron_hiperglucemia. Esto resulta lógico considerando que la presencia de hiperglucemia actúa como un indicador indirecto hacia un potencial riesgo de diabetes.

- Los resultados obtenidos revelan que varios de los modelos propuestos obtuvieron muy buenos rendimientos tanto en el problema de clasificación como de regresión para las segmentaciones DCL-bin, DCG-bin y DCL. En particular, RF, DT y ANN demostraron gran poder predictivo, con valores considerables en las métricas relevantes para cada uno de los problemas abordados. Por otro lado, el modelo LOR se situó por debajo de estos, seguido por el modelo k-NN, que exhibió el rendimiento más bajo para dichas segmentaciones.

- Las distintas segmentaciones propuestas intentan dilucidar la compensación entre la utilización de todos los *features* disponibles frente a reducir la cantidad de registros, o bien, reducir el número de variables con el objetivo de aumentar los ejemplos. En este contexto, aunque la segmentación que solo considera la información clínica (DC-bin) contenga valores fácilmente proporcionables por los pacientes, genera un impacto crítico en el rendimiento de los modelos evaluados.

Esta afirmación se respalda aún más por los resultados obtenidos durante el proceso de selección de *features*.

- A partir del proceso de selección de *features*, se destaca la importancia crítica de la `glucemia_basal` en la evaluación y detección del riesgo de diabetes. En términos generales, los modelos mantuvieron su rendimiento a medida que se eliminaron las demás variables de laboratorio. No obstante, la exclusión de la `glucemia_basal` conllevó un impacto significativo en el rendimiento de los modelos predictivos.

- Finalmente, es importante destacar la decisión de agrupar "PDM" y "DM" como una única clase, lo cual favoreció al equilibrio de las clases y simplificó el problema al volverlo de clasificación. Aunque esta simplificación conlleve el costo de no poder distinguir entre pacientes con PDM y DM, desde una perspectiva médica, esta distinción puede no ser tan grave. Lo fundamental es identificar a las personas en riesgo, independientemente de la etapa específica de la enfermedad.

Tanto los modelos predictivos a evaluar como las distintas segmentaciones generan un conjunto de posibilidades que, a priori, amplían el espectro para abordar distintas necesidades a futuro así como también hacen más enriquecedora y completa las comparaciones realizadas en la presente investigación. Resulta importante mencionar que el propósito del presente trabajo no reside en reemplazar los mecanismos de diagnóstico de la enfermedad en cuestión, sino en ayudar a identificar quienes deban realizarse las pruebas pertinentes con el objetivo de obtener un diagnóstico temprano y, por consiguiente, una intervención rápida para su control.

Debido a limitaciones propias de la base de datos, no es posible afirmar que los resultados sean concluyentes, aunque sí resultan promisorios. Considerando la vacancia de herramientas de este tipo para la población Argentina, se puede concluir que se ha cumplido con el objetivo propuesto en la presente investigación al dar el primer paso hacia el desarrollo de modelos más sofisticados en este contexto particular.

5.2 Líneas de trabajo futuras

Considerando el alcance de la presente investigación, se reconoce la existencia de ciertos aspectos que no pudieron ser abordados y que podrían resultar de interés, o al menos merecer consideración, en futuras investigaciones. Por lo tanto, se plantean las siguientes líneas de trabajo futuras para la consideración del lector:

- *Evaluación de modelos adicionales*. La evaluación de otros modelos de clasificación o regresión, como NB o SVM, que son considerados en el estado del arte, posibilita la comparación de diversos algoritmos y la identificación del que ofrezca un mejor rendimiento en la tarea de predicción del riesgo de diabetes. Esta investigación adicional es de suma importancia, ya que permite explorar y analizar las capacidades predictivas de estos modelos avanzados y determinar su idoneidad en el contexto específico del riesgo de diabetes.

- *Explorar y evaluar estrategias avanzadas*. Con el propósito de enriquecer aún más el análisis y la comparativa, es posible llevar a cabo la exploración de distintas técnicas avanzadas de ML, como el *Boosting*.

- *Búsqueda de hiperparámetros*. Con el objetivo de obtener la mejor versión de los modelos experimentados, se podrá llevar a cabo una exhaustiva búsqueda de hiperparámetros. Para ello, existen técnicas avanzadas como la optimización bayesiana, *Grid Search* y *Random Search*. Estas metodologías permiten explorar y evaluar diferentes combinaciones de hiperparámetros con el propósito de identificar aquella configuración que maximice el rendimiento de los modelos.

- *Análisis y tratamiento de valores nulos*. Es posible llevar a cabo un análisis de los valores nulos presentes en el conjunto de datos, junto con la presentación de diversas técnicas para abordar su completitud. Como se ha mencionado en la presente investigación, se puede aplicar una fórmula basada en la glucemia basal para reemplazar los valores nulos de la hemoglobina glucosilada. Además, se observa que el colesterol LDL presenta una cantidad considerable de valores faltantes. Con el propósito de abordar esta situación, se propone explorar diferentes enfoques para regresionar los valores ausentes a partir de las otras características disponibles en el conjunto de datos. Al considerar alternativas para completar los valores faltantes, se busca minimizar la pérdida de información y garantizar que los modelos de predicción del riesgo de diabetes se basen en datos confiables y completos.

- *Incremento del tamaño del conjunto de datos*. A pesar de haber obtenido modelos con rendimientos satisfactorios, existe la posibilidad de mejorar aún más su desempeño mediante el aumento del tamaño del conjunto de datos. Esta estrategia puede contribuir significativamente a mejorar el rendimiento de los modelos, particularmente en la predicción de las clases minoritarias y en la discriminación de los diferentes estadios de la diabetes. La ampliación del conjunto de datos puede ser lograda a partir de la recolección de nuevos registros de pacientes, con o sin riesgo de diabetes. Por otra parte, se pueden emplear técnicas de *data-augmentation* para generar datos adicionales a partir de los existentes en el conjunto de datos.

- *Desarrollo de una aplicación web*. A partir de los modelos obtenidos en la presente investigación, así como de aquellos generados mediante la búsqueda de hiperparámetros, se propone el desarrollo de un sitio web de acceso público. El objetivo de este sitio web es permitir a cualquier individuo proporcionar su información clínica, y opcionalmente de laboratorio, con el fin de obtener una predicción del riesgo de desarrollar diabetes basada en los modelos previamente entrenados. Asimismo, es posible considerar funcionalidades adicionales, como la visualización de resultados, recomendaciones de estilo de vida saludable o la posibilidad de mantener un seguimiento a largo plazo de los usuarios. Es importante garantizar la privacidad y seguridad de la información proporcionada por los usuarios.

Dada la naturaleza asintomática de la enfermedad, el último punto mencionado podría representar un hito significativo en la detección temprana del riesgo de diabetes. Al brindar una plataforma accesible y de fácil uso, se espera que tanto pacientes como profesionales de la salud puedan beneficiarse de esta herramienta al permitir una detección temprana y una intervención adecuada, lo que a su vez contribuiría a la prevención y el manejo eficiente de la enfermedad.

Referencias

[1] N. I. of Diabetes, Digestive, and K. Diseases, *Información general sobre la diabetes | NIDDK*, accedido: 2021-08-19. [Online]. Available: https://www.niddk.nih.gov/health-information/informacion-de-la-salud/diabetes/informacion-general.

[2] International Diabetes Federation, *Type 1 diabetes*, accedido: 2021-08-09. [Online]. Available: https://www.idf.org/aboutdiabetes/type-1-diabetes.html.

[3] International Diabetes Federation, *Type 2 diabetes*, accedido: 2021-08-09. [Online]. Available: https://www.idf.org/aboutdiabetes/type-2-diabetes.html.

[4] International Diabetes Federation, *IDF Diabetes Atlas*, 9th edn. Brussels, Belgium, 2019. [Online]. Available: https://www.diabetesatlas.org.

[5] Instituto Nacional de Estadística y Censos (INDEC) - Secretaría de Gobierno de Salud de la Nación, "4° Encuesta Nacional de Factores de Riesgo. Resultados definitivos," Buenos Aires, Tech. Rep., Oct. 2019. [Online]. Available: https://www.argentina.gob.ar/sites/default/files/sintesis-natalidad-y-mortalidad-nro6_2018-_v3.pdf.

[6] J. Rosas-Saucedo, A. E. Caballero, G. Brito-Córdova, *et al.*, "Consenso de prediabetes. documento de posición de la asociación latinoamericana de diabetes (ALAD)," *Alad*, vol. 7, no. 4, 2017. DOI: 10.24875/ALAD.17000307.

[7] International Diabetes Federation, *Complications*, accedido: 2021-08-09. [Online]. Available: https://www.idf.org/aboutdiabetes/complications.html.

[8] J. J. Gagliardino, D. Assad, G. G. Gagliardino, *et al.*, *Cómo tratar mi diabetes*, 3rd ed. Buenos Aires, Argentina, 2016.

[9] World Health Organization, *Global action plan for the prevention and control of noncommunicable diseases, 2013-2020*. World Health Organization, 2013. [Online]. Available: https://apps.who.int/iris/handle/10665/94384.

[10] International Diabetes Federation, *IDF Diabetes Atlas*, 7th edn. Brussels, Belgium, 2015. [Online]. Available: https://www.diabetesatlas.org.

[11] *En Argentina el 10 por ciento de la población sufre de diabetes Tipo 1*, 2021. [Online]. Available: https://www.lavoz.com.ar/ciudadanos/en-argentina-el-10-por-ciento-de-la-poblacion-sufre-de-diabetes-tipo-1/.

[12] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959. DOI: 10.1147/rd.33.0210.

[13] A. Conner-Simons, *Using artificial intelligence to improve early breast cancer detection*, accedido: 2021-08-08, 2017. [Online]. Available: https://news.mit.edu/2017/artificial-intelligence-early-breast-cancer-detection-1017.

[14] D. S. Kermany, M. Goldbaum, W. Cai, *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, 1122–1131.e9, 2018. DOI: 10.1016/j.cell.2018.02.010.

[15] C. J. Gunasekara, E. Hannon, H. MacKay, *et al.*, "A machine learning case–control classifier for schizophrenia based on DNA methylation in blood," *Translational Psychiatry*, vol. 11, no. 1, 2021. DOI: 10.1038/s41398-021-01496-3.

[16] S. Qiu, P. S. Joshi, M. I. Miller, *et al.*, "Development and validation of an interpretable deep learning framework for Alzheimer's disease classification," *Brain: a journal of neurology*, vol. 143, no. 6, pp. 1920–1933, 2020. DOI: 10.1093/brain/awaa137.

[17] *CENEXA | Programa de Prevención Primaria de la Diabetes (PPDBA)*, accedido: 2021-08-10. [Online]. Available: https://www.ppdba.cenexa.org/.

[18] J. J. Gagliardino, G. Etchegoyen, M. Bourgeois, *et al.*, "Prevención primaria de diabetes tipo 2 en Argentina: Estudio piloto en la provincia de Buenos Aires," *Revista Argentina de Endocrinología y Metabolismo*, vol. 53, no. 4, pp. 135–141, Oct. 2016. DOI: 10.1016/j.raem.2016.11.002.

[19] J. Lindström and J. Tuomilehto, "The diabetes risk score: A practical tool to predict type 2 diabetes risk," *Diabetes care*, vol. 26, no. 3, pp. 725–731, 2003. DOI: 10.2337/diacare.26.3.725.

[20] *Pandas - Python Data Analysis Library*, accedido: 2021-08-12. [Online]. Available: https://pandas.pydata.org/.

[21] W. McKinney, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.

[22] C. R. Harris, K. J. Millman, S. J. van der Walt, *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020. DOI: 10.1038/s41586-020-2649-2.

[23] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. DOI: 10.1109/MCSE.2007.55.

[24] *Tensorflow*, accedido: 2021-08-12. [Online]. Available: https://www.tensorflow.org/?hl=es-419.

[25] *Scikit-learn*, accedido: 2021-08-12. [Online]. Available: https://scikit-learn.org/stable/.

[26] N. D. D. I. Clearinghouse, *El aparato digestivo y su funcionamiento*, accedido: 2022-01-24, 2005. [Online]. Available: https://web.archive.org/web/20051029025618/http://digestive.niddk.nih.gov/spanish/pubs/yrdd/index.htm.

[27] Mediversia, *DIABETES MELLITUS - tipo 1 y 2, fisiopatología, cetoacidosis diabética, diagnóstico y tratamiento*, accedido: 2022-02-23. [Online]. Available: https://www.youtube.com/watch?v=PPbrfPs53vM.

[28] J. Pérez Porto and M. Merino, *Glucógeno - qué es, definición y concepto*, accedido: 2022-01-24, 2018. [Online]. Available: https://definicion.de/glucogeno/.

[29] U. F. de Ciencias. Nutrición y Dietética., *Guía de alimentación y salud UNED: Guía de nutrición > la composición de los alimentos > hidratos de carbono*, accedido: 2022-01-24. [Online]. Available: https://www2.uned.es/pea-nutricion-y-dietetica-I/guia/guia_nutricion/compo_hidratos.htm.

[30] Facultad de Ciencias Médicas, *FISIOLOGÍA: Páncreas endocrino. PARTE i*, accedido: 2022-01-24. [Online]. Available: https://www.youtube.com/watch?v=K4sGGwwaLxk.

[31] F. E. de Diabetes FEDE, *Diabetes e insulina*, accedido: 2022-01-24. [Online]. Available: https://fedesp.es/diabetes/insulina/.

[32] N. I. of Diabetes, Digestive, and K. Diseases, *¿qué es la diabetes? | NIDDK*, accedido: 2021-12-17. [Online]. Available: https://www.niddk.nih.gov/health-information/informacion-de-la-salud/diabetes/informacion-general/que-es.

[33] L. Howells, B. Musaddaq, A. J. McKay, and A. Majeed, "Clinical impact of lifestyle interventions for the prevention of diabetes: An overview of systematic reviews," *BMJ open*, vol. 6, no. 12, e013806, 2016. DOI: 10.1136/bmjopen-2016-013806.

[34] World Health Organization, *Classification of diabetes mellitus*. World Health Organization, 2019. [Online]. Available: https://apps.who.int/iris/handle/10665/325182.

[35] L. L. J. Koppes, J. M. Dekker, H. F. J. Hendriks, L. M. Bouter, and R. J. Heine, "Moderate alcohol consumption lowers the risk of type 2 diabetes: A meta-analysis of prospective observational studies," *Diabetes Care*, vol. 28, no. 3, pp. 719–725, 2005. DOI: 10.2337/diacare.28.3.719.

[36] S. Carlsson, N. Hammar, and V. Grill, "Alcohol consumption and type 2 diabetes meta-analysis of epidemiological studies indicates a u-shaped relationship," *Diabetologia*, vol. 48, no. 6, pp. 1051–1054, 2005. DOI: 10.1007/s00125-005-1768-5.

[37] C. Willi, P. Bodenmann, W. A. Ghali, P. D. Faris, and J. Cornuz, "Active smoking and the risk of type 2 diabetes: A systematic review and meta-analysis," *JAMA*, vol. 298, no. 22, pp. 2654–2664, 2007. DOI: 10.1001/jama.298.22.2654.

[38] J. Gangwisch, S. Heymsfield, B. Boden-Albala, *et al.*, "Sleep duration as a risk factor for diabetes incidence in a large u.s. sample," *Sleep*, vol. 30, pp. 1667–73, 2007. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2276127/.

[39] C.-Y. Chao, J.-S. Wu, Y.-C. Yang, *et al.*, "Sleep duration is a potential risk factor for newly diagnosed type 2 diabetes mellitus," *Metabolism: Clinical and Experimental*, vol. 60, no. 6, pp. 799–804, 2011. DOI: 10.1016/j.metabol.2010.07.031.

[40] W. H. Organization, *Diagnosis and management of type 2 diabetes (HEARTS-D)*. World Health Organization, 2020. [Online]. Available: https://www.who.int/publications/i/item/who-ucn-ncd-20.1.

[41] N. I. of Diabetes, Digestive, and K. Diseases, *Síntomas y causas de la diabetes | NIDDK*, accedido: 2021-12-18. [Online]. Available: https://www.niddk.nih.gov/health-information/informacion-de-la-salud/diabetes/informacion-general/sintomas-causas.

[42] W. H. Organization, *Diabetes*. [Online]. Available: https://www.who.int/es/news-room/fact-sheets/detail/diabetes.

[43] N. I. of Diabetes, Digestive, and K. Diseases, *Pruebas y diagnóstico de la diabetes | NIDDK*, accedido: 2021-12-18. [Online]. Available: https://www.niddk.nih.gov/health-information/informacion-de-la-salud/diabetes/informacion-general/pruebas-diagnostico.

[44] *Glucemia capilar*, accedido: 2021-12-18. [Online]. Available: https://www.sanitas.es/sanitas/seguros/es/particulares/biblioteca-de-salud/diabetes/glucemia-capilar.html.

[45] International Diabetes Federation, *Guía de incidencia política*, 2019. [Online]. Available: https://diabetesatlas.org/upload/resources/material/20191219_091956_2019_IDF_Advocacy_Guide_ES.pdf.

[46] A. Kharroubi and H. Darwish, "Diabetes mellitus: The epidemic of the century," *World Journal of Diabetes*, vol. 6, no. 6, pp. 850–867, 2015. DOI: 10.4239/wjd.v6.i6.850.

[47] Federación Internacional de Diabetes, *América del sur y central*, Annual Report, 2019, pp. 72–73. [Online]. Available: https://www.diabetesatlas.org/upload/resources/material/20200302_133352_2406-IDF-ATLAS-SPAN-BOOK.pdf.

[48] Organización Panamericana de la Salud and Organización Mundial de la Salud, "Agenda de salud sostenible para las américas 2018-2030: Un llamado a la acción para la salud y el bienestar en la región," Tech. Rep., 2017. [Online]. Available: https://iris.paho.org/handle/10665.2/49169.

[49] Ministerio de Salud Argentina, "Natalidad y mortalidad 2018," Tech. Rep., 2018, p. 20. [Online]. Available: https://www.argentina.gob.ar/sites/default/files/sintesis-natalidad-y-mortalidad-nro6_2018-_v3.pdf.

[50] M. de Salud de la República Argentina, *Diabetes mellitus*, accedido: 2022-02-23, 2017. [Online]. Available: https://www.argentina.gob.ar/salud/glosario/diabetes.

[51] Asociación Latinoamericana de Diabetes, *Guías ALAD sobre el diagnóstico, control y tratamiento de la diabetes mellitus tipo 2 con medicina basada en evidencia*, 2019. [Online]. Available: https://www.revistaalad.com/guias/5600AX191_guias_alad_2019.pdf.

[52] N. I. of Diabetes, Digestive, and K. Diseases, *Cómo prevenir la diabetes tipo 2 | NIDDK*, accedido: 2022-03-19. [Online]. Available: https://www.niddk.nih.gov/health-information/informacion-de-la-salud/diabetes/informacion-general/prevenir-tipo-2.

[53] S. Murillo, *Prevención de la diabetes tipo 2 mediante alimentación y ejercicio*, accedido: 2022-03-19, 2011. [Online]. Available: https://www.fundaciondiabetes.org/general/articulo/57/prevencion-de-la-diabetes-tipo-2-mediante-alimentacion-y-ejercicio.

[54] D. Zhang, S. Mishra, E. Brynjolfsson, *et al.*, "The AI index 2021 annual report," Tech. Rep., 2021. [Online]. Available: https://aiindex.stanford.edu/wp-content/uploads/2021/11/2021-AI-Index-Report_Master.pdf.

[55] S. R. Choudhury, *Google DeepMind's AlphaGo beats go champion lee sedol in AI milestone in seoul*, accedido: 2022-02-16, 2016. [Online]. Available: https://www.cnbc.com/2016/03/08/google-deepminds-alphago-takes-on-go-champion-lee-sedol-in-ai-milestone-in-seoul.html.

[56] S. Shead, *Google DeepMind's AI can detect 50 eye disease conditions and save sight*, accedido: 2022-02-16, 2018. [Online]. Available: https://www.forbes.com/sites/samshead/2018/08/13/google-deepminds-ai-can-detect-50-eye-disease-conditions-and-save-sight/.

[57] M. V. S. Nadal, *Un sistema de inteligencia artificial fue el primero en alertar del coronavirus de Wuhan*, accedido: 2022-02-16, 2020. [Online]. Available: https://elpais.com/tecnologia/2020/01/28/actualidad/1580235041_105388.html.

[58] D. Poole, A. Mackworth, and R. Goebel, *Computational Intelligence: A Logical Approach*. 1998. [Online]. Available: https://www.cs.ubc.ca/~poole/ci.html.

[59] A. Kaplan and M. Haenlein, "Siri, siri, in my hand: Who's the fairest in the land? on the interpretations, illustrations, and implications of artificial intelligence," *Business Horizons*, vol. 62, no. 1, pp. 15–25, 2019. DOI: 10.1016/j.bushor.2018.08.004.

[60] A. M. Turing, "Computing machinery and intelligence," *Mind, New Series*, vol. 59, no. 236, pp. 433–460, 1950. [Online]. Available: http://www.jstor.org/stable/2251299.

[61] F. Chollet, *Deep Learning with Python*. Manning, 2017, ISBN: 978-1-61729-443-3.

[62] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org.

[63] T. M. Mitchell, *Machine Learning*, ser. McGraw-Hill series in computer science. McGraw-Hill Education, 1997, ISBN: 978-0-07-042807-2.

[64] Jason Bell, *Machine Learning: Hands-on for Developers and Technical Professionals*. Wiley, 2014, ISBN: 978-1-118-88906-0.

[65] S. Brown, *Machine learning, explained*, accedido: 2022-02-16, 2021. [Online]. Available: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained.

[66] Andreas C. Müller and Sarah Guido, *Introduction to Machine Learning with Python*. O'Reilly Media, Inc., 2017, ISBN: 978-1-4493-6941-5.

[67] *Siri*, accedido: 2022-03-19. [Online]. Available: https://www.apple.com/es/siri/.

[68] *Amazon Alexa Official Site: What is Alexa?* accedido: 2022-03-19. [Online]. Available: https://developer.amazon.com/en-US/alexa.

[69] *Google Assistant, your own personal Google*, accedido: 2022-03-19. [Online]. Available: https://assistant.google.com/.

[70] D. Yu, *Parking lot vehicle detection using deep learning*, accedido: 2022-02-16, 2019. [Online]. Available: https://medium.com/geoai/parking-lot-vehicle-detection-using-deep-learning-49597917bc4a.

[71] R. Gordon, *Robust artificial intelligence tools to predict future cancer*, accedido: 2022-02-16, 2021. [Online]. Available: https://news.mit.edu/2021/robust-artificial-intelligence-tools-predict-future-cancer-0128.

[72] E. Ackerman, *How drive.ai is mastering autonomous driving with deep learning*, accedido: 2022-02-16, 2017. [Online]. Available: https://spectrum.ieee.org/how-driveai-is-mastering-autonomous-driving-with-deep-learning.

[73] *Netflix*, accedido: 2022-03-19. [Online]. Available: https://www.netflix.com/.

[74] *Youtube*, accedido: 2022-03-19. [Online]. Available: https://www.youtube.com/.

[75] *Facebook*, accedido: 2022-03-19. [Online]. Available: https://es-la.facebook.com/.

[76] *Instagram*, accedido: 2022-03-19. [Online]. Available: https://instagram.com/.

[77] *Amazon*, accedido: 2022-03-19. [Online]. Available: https://www.amazon.com/.

[78] IBM, *Cancer research and treatment | IBM*, accedido: 2022-02-16. [Online]. Available: https://www.ibm.com/watson-health/solutions/cancer-research-treatment.

[79] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, *Learning from data*. Departments of Electrical Engineering and Computer Science, California Institute of Technology, 2012, ISBN: 1-60049-006-9.

[80] S. Marsland, *Machine Learning: An Algorithmic Perspective*, ser. Second Edition. Chapman and Hall/CRC, 2014, ISBN: 978-1-4665-8333-7.

[81] Stuart J. Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 3rd. Upper Saddle River: Prentice Hall Press, 2009, ISBN: 978-0-13-604259-4.

[82] C. Schröer, F. Kruse, and J. Marx Gómez, "A systematic literature review on applying crisp-dm process model," *Procedia Computer Science*, vol. 181, pp. 526–534, Jan. 2021. DOI: 10.1016/j.procs.2021.01.199.

[83] O. Marbán, G. Mariscal, and J. Segovia, "A data mining & knowledge discovery process model," in *Data Mining and Knowledge Discovery in Real Life Applications*. Rijeka, Croacia, Jan. 2009, pp. 1–17. DOI: 10.5772/6438.

[84] P. Chapman, J. Clinton, R. Kerber, *et al.*, "CRISP-DM 1.0: Step-by-step data mining guide," 2000.

[85] D. Sarkar, R. Bali, and T. Sharma, *Practical Machine Learning with Python*. Apress Berkeley, CA, 2018, ISBN: 978-1-4842-3207-1. [Online]. Available: http://link.springer.com/10.1007/978-1-4842-3207-1.

[86] A. Mavuduru, *Is Data Really the New Oil in the 21st Century?* accedido: 2022-07-10, 2020. [Online]. Available: https://towardsdatascience.com/is-data-really-the-new-oil-in-the-21st-century-17d014811b88.

[87] J. F. Hair JR, J. B. Babin, R. E. Anderson, and W. C. Black, *Multivariate data analysis*, 7th ed. Pearson, 2009, ISBN: 978-0-13-813263-7.

[88] B. Suthar, H. Patel, A. Goswami, and M. tech. Scholar, "A survey: Classification of imputation methods in data mining," 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:2058666.

[89] R. Houari, A. Bounceur, A. Tari, and M. Kecha, "Handling missing data problems with sampling methods," Jun. 2014, pp. 99–104. DOI: 10.1109/INDS.2014.25.

[90] H. Kang, "The prevention and handling of the missing data," *Korean Journal of Anesthesiology*, vol. 64, no. 5, pp. 402–406, May 2013. DOI: 10.4097/kjae.2013.64.5.402.

[91] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *Journal of Big Data*, vol. 8, no. 1, p. 140, Oct. 27, 2021. DOI: 10.1186/s40537-021-00516-9.

[92] T. Nasima, *Tackling Missing Value in Dataset*, accedido: 2022-07-10, 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/10/handling-missing-value/.

[93] R. J. A. Little and D. B. Rubin, *Statistical analysis with missing data*, Third edition, ser. Wiley series in probability and statistics. Wiley, 2020, ISBN: 978-0-470-52679-8.

[94] P. E. McKnight, K. M. McKnight, S. Sidani, and A. J. Figueredo, *Missing data: a gentle introduction*, ser. Methodology in the social sciences. Guilford Press, 2007, ISBN: 978-1-59385-393-8.

[95] A. R. T. Donders, G. J. M. G. v. d. Heijden, T. Stijnen, and K. G. M. Moons, "Review: A gentle introduction to imputation of missing values," *Journal of Clinical Epidemiology*, vol. 59, no. 10, pp. 1087–1091, Oct. 2006. DOI: 10.1016/j.jclinepi.2006.01.014.

[96] P. J. García-Laencina, J.-L. Sancho-Gómez, A. R. Figueiras-Vidal, and M. Verleysen, "K nearest neighbours with mutual information for simultaneous classification and missing data imputation," *Neurocomputing*, vol. 72, no. 7, pp. 1483–1493, 2009. DOI: 10.1016/j.neucom.2008.11.026.

[97] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *Journal of Big Data*, vol. 8, no. 1, p. 140, Oct. 2021. DOI: 10.1186/s40537-021-00516-9.

[98] C. C. Aggarwal, *Outlier Analysis*, 2nd ed. Springer Cham, 2016, ISBN: 978-3-319-47577-6. [Online]. Available: http://link.springer.com/10.1007/978-3-319-47578-3.

[99] J. W. Tukey, *Exploratory data analysis*, ser. Addison-Wesley series in behavioral science. Reading, Mass: Addison-Wesley Pub. Co, 1977, ISBN: 978-0-387-32833-1.

[100] R. Peck, C. Olsen, and J. L. Devore, *Introduction to statistics and data analysis*, 3rd ed. Cengage Learning, 2008, ISBN: 978-0-495-11873-2.

[101] M. N. Bernstein, *Intrinsic dimensionality*, accedido: 2022-07-11, 2020. [Online]. Available: https://mbernste.github.io/posts/intrinsic_dimensionality/.

[102] E. Schubert and M. Gertz, "Intrinsic t-stochastic neighbor embedding for visualization and outlier detection," Oct. 2017, pp. 188–203, ISBN: 978-3-319-68474-1. DOI: 10.1007/978-3-319-68474-1_13.

[103] M. Bermingham, R. Pong-Wong, A. Spiliopoulou, *et al.*, "Application of high-dimensional feature selection: Evaluation for genomic prediction in man," *Scientific Reports*, vol. 5, p. 10 312, 2015. DOI: 10.1038/srep10312.

[104] J. A. Rodrigo, *PCA con Python*, accedido: 2022-07-11, 2020. [Online]. Available: https://www.cienciadedatos.net/documentos/py19-pca-python.html.

[105] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," ser. NIPS'04, Vancouver, British Columbia, Canada: MIT Press, 2004, pp. 513–520. [Online]. Available: https://www.cs.toronto.edu/~hinton/absps/nca.pdf.

[106] Scikit-Learn, *Dimensionality Reduction with Neighborhood Components Analysis*, accedido: 2023-07-08. [Online]. Available: https://scikit-learn.org/stable/auto_examples/neighbors/plot_nca_dim_reduction.html.

[107] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936. DOI: 10.1111/j.1469-1809.1936.tb02137.x.

[108] A. Martinez and A. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, Feb. 2001. DOI: 10.1109/34.908974.

[109] C. Lemaréchal, "Cauchy and the gradient method," in *Optimization Stories*, M. Grötschel, Ed., 1st ed., EMS Press, Jan. 1, 2012, pp. 251–254. DOI: 10.4171/dms/6/27.

[110] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, Mar. 1951. DOI: 10.1214/aoms/1177729694.

[111] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948. [Online]. Available: https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf.

[112] Scikit-Learn, *1.11. Ensemble methods*, accedido: 2023-07-08. [Online]. Available: https://scikit-learn.org/stable/modules/ensemble.html.

[113] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996. DOI: 10.1007/BF00058655.

[114] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997. DOI: 10.1006/jcss.1997.1504.

[115] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Aug. 2016, pp. 785–794. DOI: 10.1145/2939672.2939785.

[116] S. Raschka, *STAT 479: Machine Learning Lecture Notes*, University of Wisconsin–Madison, 2018. [Online]. Available: https://pages.stat.wisc.edu/~sraschka/teaching/stat479-fs2019/.

[117] IBM, *What is the k-nearest neighbors algorithm? | IBM*, accedido: 2022-07-11. [Online]. Available: https://www.ibm.com/topics/knn.

[118] E. Brian S., L. Sabine, L. Morven, and S. Daniel, *Cluster analysis*, 5th ed, ser. Wiley series in probability and statistics. Chichester, West Sussex, U.K: Wiley, Jan. 2011, ISBN: 978-0-470-74991-3.

[119] Scikit-Learn, *1.6. Nearest Neighbors*, accedido: 2023-07-08. [Online]. Available: https://scikit-learn/stable/modules/neighbors.html.

[120] M. Minsky and S. A. Papert, *Perceptrons: An Introduction to Computational Geometry*. Cambridge/Mass.: The MIT Press, Dec. 1972, ISBN: 978-0-262-34393-0. DOI: 10.7551/mitpress/11301.001.0001.

[121] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, Dec. 1943. DOI: 10.1007/BF02478259.

[122] M. A. Nielsen, "Neural Networks and Deep Learning," *Determination Press*, 2015.

[123] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain.,” *Psychological review*, vol. 65 6, pp. 386–408, 1958. [Online]. Available: https://api.semanticscholar.org/CorpusID:12781225.

[124] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Delhi: Prentice Hall, Jan. 1998, ISBN: 978-0132733502.

[125] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989. DOI: 10.1016/0893-6080(89)90020-8.

[126] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, Dec. 1, 1989. DOI: 10.1007/BF02551274.

[127] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, *Activation Functions: Comparison of trends in Practice and Research for Deep Learning*, Nov. 2018. DOI: 10.48550/arXiv.1811.03378.

[128] Y. Goldberg, *Neural network methods for natural language processing*, ser. Synthesis lectures on human language technologies. San Rafael, Calif.: Morgan & Claypool Publishers, 2017, ISBN: 978-1-62705-298-6.

[129] D. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, pp. 533–536, 1986. DOI: 10.1038/323533a0.

[130] H. Anton, *Calculus: A New Horizon*, 6th ed. New York: John Wiley & Sons, Inc., 1998, ISBN: 9780471316763.

[131] D. E. Rumelhart, J. L. McClelland, and PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*. The MIT Press, Jul. 17, 1986, ISBN: 978-0-262-29140-8. DOI: 10.7551/mitpress/5236.001.0001.

[132] M. V. Valueva, N. N. Nagornov, P. A. Lyakhov, G. V. Valuev, and N. I. Chervyakov, “Application of the residue number system to reduce hardware costs of the convolutional neural network implementation,” *Mathematics and Computers in Simulation*, vol. 177, pp. 232–243, 2020. DOI: 10.1016/j.matcom.2020.04.031.

[133] L. Bottou and O. Bousquet, “The tradeoffs of large scale learning,” in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, Curran Associates, Inc., 2007. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2007/file/0d3180d672e08b4c5312dcdafdf6ef36-Paper.pdf.

[134] N. Buduma and N. Locascio, *Fundamentals of Deep Learning: designing next-generation machine intelligence algorithms*. "O’Reilly Media, Inc.", May 2017, ISBN: 978-1-4919-2561-4.

[135] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014. [Online]. Available: https://www.cs.toronto.edu/~hinton/absps/JMLRdropout.pdf.

[136] J. A. Cruz and D. S. Wishart, "Applications of machine learning in cancer prediction and prognosis," *Cancer Informatics*, vol. 2, 2006. DOI: 10.1177/117693510600200030.

[137] F. Maleki, K. Ovens, K. Najafian, B. Forghani, C. Reinhold, and R. Forghani, "Overview of machine learning part 1: Fundamentals and classic approaches," *Neuroimaging Clinics of North America*, vol. 30, no. 4, e17–e32, 2020. DOI: 10.1016/j.nic.2020.08.007.

[138] Y. M. Haibo He, *Imbalanced learning: foundations, algorithms, and applications*, H. He and Y. Ma, Eds. Hoboken, New Jersey: John Wiley & Sons, Inc, 2013, ISBN: 978-1-118-64633-5.

[139] F. Alberto, G. Salvador, G. Mikel, P. Ronaldo C., K. Bartosz, and H. Francisco, *Learning from Imbalanced Data Sets*. Springer Cham, 2018, ISBN: 978-3-319-98074-4. DOI: 10.1007/978-3-319-98074-4.

[140] P. Branco, L. Torgo, and R. Ribeiro, *A survey of predictive modelling under imbalanced distributions*, May 2015. DOI: 10.48550/arXiv.1505.01658.

[141] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated Machine Learning: Methods, Systems, Challenges*, F. Hutter, L. Kotthoff, and J. Vanschoren, Eds., Cham: Springer International Publishing, 2019, pp. 3–33, ISBN: 978-3-030-05318-5. DOI: 10.1007/978-3-030-05318-5_1.

[142] A. Mir and S. N. Dhage, "Diabetes disease prediction using machine learning on big data of healthcare," 2018, pp. 1–6. DOI: 10.1109/ICCUBEA.2018.8697439.

[143] M. Kahn, *UCI Machine Learning Repository: Diabetes Data Set*, accedido: 2021-08-08, 1994. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/diabetes.

[144] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang, "Predicting Diabetes Mellitus With Machine Learning Techniques," *Frontiers in Genetics*, vol. 9, p. 515, Nov. 2018. DOI: 10.3389/fgene.2018.00515.

[145] W. Yu, T. Liu, R. Valdez, M. Gwinn, and M. J. Khoury, "Application of support vector machine modeling for prediction of common diseases: The case of diabetes and pre-diabetes," *BMC Medical Informatics and Decision Making*, vol. 10, no. 1, p. 16, Mar. 2010. DOI: 10.1186/1472-6947-10-16.

[146] X.-H. Meng, Y.-X. Huang, D.-P. Rao, Q. Zhang, and Q. Liu, "Comparison of three data mining models for predicting diabetes or prediabetes by risk factors," *The Kaohsiung Journal of Medical Sciences*, vol. 29, no. 2, pp. 93–99, 2013. DOI: 10.1016/j.kjms.2012.08.016.

[147] S. B. Choi, W. J. Kim, T. K. Yoo, *et al.*, "Screening for Prediabetes Using Machine Learning Models," *Computational and Mathematical Methods in Medicine*, vol. 2014, Jul. 2014, Publisher: Hindawi. DOI: 10.1155/2014/618976.

[148] A. R. Olivera, V. Roesler, C. Iochpe, *et al.*, "Comparison of machine-learning algorithms to build a predictive model for detecting undiagnosed diabetes - ELSA-Brasil: Accuracy study," *Sao Paulo Medical Journal*, vol. 135, pp. 234–246, Jun. 2017, Publisher: Associação Paulista de Medicina - APM. DOI: 10.1590/1516-3180.2016.0309010217.

[149] CENEXA, *Programa Prevención Primaria de Diabetes Tipo 2 (PPDBA). Cuestionario de riesgo de Diabetes Tipo 2*, accedido: 2022-10-30, 2014. [Online]. Available: https://www.ppdba.cenexa.org/uploads/assets/Cuestionario_de_Riesgo.pdf.

[150] A. H. Association, *What Your Cholesterol Levels Mean*, accedido: 2022-10-10, 2020. [Online]. Available: https://www.heart.org/en/health-topics/cholesterol/about-cholesterol/what-your-cholesterol-levels-mean.

[151] C. L. Rohlfing, H.-M. Wiedmeyer, R. R. Little, J. D. England, A. Tennill, and D. E. Goldstein, "Defining the Relationship Between Plasma Glucose and HbA1c: Analysis of glucose profiles and HbA1c in the Diabetes Control and Complications Trial," *Diabetes Care*, vol. 25, no. 2, pp. 275–278, Feb. 2002. DOI: 10.2337/diacare.25.2.275.

[152] Scikit-Learn, *3.1. Cross-validation: Evaluating estimator performance*, accedido: 2023-04-11. [Online]. Available: https://scikit-learn.org/stable/modules/cross_validation.html.

[153] Jupyter Notebook Documentation, *Jupyter Notebook 7.0.0rc2 documentation*, accedido: 2023-04-18. [Online]. Available: https://jupyter-notebook.readthedocs.io/en/latest/index.html.

[154] F. para la Diabetes, *¿qué es la diabetes?* accedido: 2021-08-13. [Online]. Available: https://www.fundaciondiabetes.org/general/82/conozcamosla-mejor.

[155] "Professional practice committee: Standards of medical care in diabetes—2018," *Diabetes Care*, vol. 41, no. 1, S3–S3, 2018. DOI: 10.2337/dc18-Sppc01.

[156] O. P. de la Salud, *Diabetes - OPS/OMS | Organización Panamericana de la Salud*, accedido: 2021-08-13. [Online]. Available: https://www.paho.org/es/temas/diabetes.

[157] F. A. de Diabetes, *¿Qué es la Diabetes? – Federación Argentina de Diabetes*, accedido: 2021-08-13, 2019. [Online]. Available: https://www.fad.org.ar/que-es-la-diabetes/.

[158] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*, Second edition. Beijing Boston Farnham Sebastopol Tokyo: O'Reilly, 2019, ISBN: 978-1-4920-3261-8.

[159] Andrew Yan-Tak Ng, *Machine learning yearning: Technical strategy for ai engineers in the era of deep learning*, 2019. [Online]. Available: https://www.mlyearning.org/.

[160] M. Bergman, M. Manco, G. Sesti, *et al.*, "Petition to replace current OGTT criteria for diagnosing prediabetes with the 1-hour post-load plasma glucose ≥ 155 mg/dl (8.6 mmol/l)," *Diabetes Research and Clinical Practice*, vol. 146, 2018. DOI: 10.1016/j.diabres.2018.09.017.

[161] W. H. Organization, *WHO launches a chatbot on Facebook Messenger to combat COVID-19 misinformation*, accedido: 2022-03-19, 2020. [Online]. Available: https://www.who.int/news-room/feature-stories/detail/who-launches-a-chatbot-powered-facebook-messenger-to-combat-covid-19-misinformation.

[162] KLM, *KLM introduces Messenger chatbot for ticket booking | Mobile Marketing Magazine*, accedido: 2022-03-19, 2017. [Online]. Available: http://mobilemarketingmagazine.com/klm-bluebot-bb-facebook-messenger-chatbot.

[163] *Banco Santander*, accedido: 2022-03-19. [Online]. Available: https://www.santander.com.ar/banco/online/personas.

[164] R. Matheson, *Reducing false positives in credit card fraud detection*, accedido: 2022-02-16, 2018. [Online]. Available: https://news.mit.edu/2018/machine-learning-financial-credit-card-fraud-0920.

[165] Q. Song and M. Shepperd, "Missing data imputation techniques," *International Journal of Business Intelligence and Data Mining*, vol. 2, no. 3, pp. 261–291, 2007. DOI: 10.1504/IJBIDM.2007.015485.

[166] R. Bellman, *Dynamic programming*, Dover ed. Mineola, N.Y: Dover Publications, 2003, ISBN: 978-0-486-42809-3.

[167] R. Bellman, *Adaptive control processes: a guided tour*, ser. Princeton Legacy Library. Princeton University Press, 1961, ISBN: 978-1-4008-7466-8. DOI: 10.1515/9781400874668.

[168] Scikit-Learn, *1.10. Decision Trees*, accedido: 2023-07-08. [Online]. Available: https://scikit-learn.org/stable/modules/tree.html.

[169] L. Breiman, J. Friedman, C. Stone, and R. Olshen, *Classification and Regression Trees*. Taylor & Francis, 1984. DOI: 10.1002/cyto.990080516.

[170] "The relationship between precision-recall and roc curves," vol. 06, Jun. 2006. DOI: 10.1145/1143844.1143874.

[171] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research 13*, vol. 1, pp. 281–305, 2012. [Online]. Available: https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf.

[172] C.-w. Hsu, C.-c. Chang, and C.-J. Lin, "A practical guide to support vector classification chih-wei hsu, chih-chung chang, and chih-jen lin," Nov. 2003.