



Universidad Politécnica
de Madrid



**Escuela Técnica Superior de
Ingenieros Informáticos**

Grado en Ciencia de Datos e Inteligencia Artificial

Trabajo Fin de Grado

**Generación de un Metamodelo Basado
en Machine Learning para Determinar
Casos de Reposicionamiento de
Fármacos**

Autor: Alba Sanz Bustos

Tutor: Alejandro Rodríguez González

Co-tutora: Lucía Prieto Santamaría

Madrid, Junio 2025

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado

Grado en Ciencia de Datos e Inteligencia Artificial

Título: Generación de un Metamodelo basado en Machine Learning para
Determinar Casos de Reposicionamiento de Fármacos
Junio, 2025

Autor: Alba Sanz Bustos

Tutor:

Alejandro Rodríguez González
Departamento de Lenguajes y Sistemas Informáticos e Ingeniería de
Software
ETSI Informáticos
Universidad Politécnica de Madrid

Co-tutora:

Lucía Prieto Santamaría
Departamento de Lenguajes y Sistemas Informáticos e Ingeniería de
Software
ETSI Informáticos
Universidad Politécnica de Madrid

Agradecimientos

Después de cuatro años de esfuerzo y sacrificio, con tantos buenos (y no tan buenos) momentos, toca decir adiós a esta etapa supongo...

Me gustaría agradecer en primer lugar a mi tutora, Lucía. Gracias por haberme ayudado durante estos meses, por haber hecho esto más “sencillo” y, sobre todo, por transmitir tanto amor hacia tu trabajo. Gracias también a Alejandro, por haberme dado la oportunidad de aprender con vosotros.

A nivel personal, podría no terminar nunca. Gracias a mis amigos de la uni, por todas las risas, viajes, llantos y horas sin dormir antes de los exámenes. Gracias por haber hecho las (pocas) horas de clase menos aburridas. Gracias Carlos por abrirme las puertas de tu casa desde el primer día, incluso sin entenderme muy bien cuando hablaba. Gracias Dani por ser tú. Sé que esto es para siempre.

Gracias a mis antiguas compis de piso (porque ahora ya son amigas) por haber convivido conmigo durante algunos años de esta etapa tan bonita y complicada a la vez. Gracias por haberme cuidado en mis peores momentos chicas. Sin vosotras, probablemente estaría desnutrida. Gracias Eva, por haber sido mi madre, hermana y amiga a la vez. Gracias.

Gracias a mis niñas de Almu, porque, a pesar de los kilómetros, nunca me han hecho sentir sola. Gracias Lydia, por ser mi hermana y llevar a mi lado tantos años, jamás me he imaginado cerrar una etapa de mi vida sin ti. Gracias Clara, por haber sido mi diario todo este tiempo, sin ti, nada de esto hubiese sido posible. Gracias Ana, por ser hogar a tantos kilómetros de casa. Os quiero.

Gracias a mi familia, por recibirme siempre que volvía a casa de la mejor manera. En especial, gracias a mis hermanas, Ana y María, Madrid con vosotras cerca es un poquito más andaluz. Gracias por haber sido mis ejemplos a seguir, habéis hecho muy bien el papel de hermanas mayores.

Gracias a mis padres, por ser todo para mí. Gracias por haber confiado siempre en mí y no dejarme nunca sola. Gracias por vuestro amor incondicional, por dar todo por vuestras hijas. Lo habéis hecho genial.

Papá, gracias por dejarme ser tu princesa.
Mamá, gracias por ser mi luz diaria.
Os quiero.

Y por último, gracias a mi misma, por no rendirme nunca.

Gracias.

Resumen

El desarrollo tradicional de fármacos exige inversiones millonarias, más de una década de trabajo e implica elevadas tasas de fracaso, de ahí el creciente interés por el reposicionamiento de fármacos, estrategia que busca asignar nuevas indicaciones terapéuticas a medicamentos ya aprobados. Sin embargo, la diversidad de modelos computacionales (basados en grafos, proximidad en redes biomoleculares o rutas multiescala) genera resultados independientes que dificultan la selección de candidatos con mayor probabilidad de éxito clínico. Este Trabajo Fin de Grado aborda esa brecha mediante el diseño y la implementación de un metamodelo de *machine learning*, que integra y sintetiza la información procedente de tres familias de modelos almacenados en la base de datos del proyecto DRIVE (incluyendo hipótesis de reposicionamiento generadas mediante *Graph Neural Networks*, *Network Proximity* e *Information Paths*).

Tras una revisión exhaustiva del estado del arte, se ha construido un conjunto de datos unificado (con más de cien mil pares enfermedad-fármaco y 12 métricas predictoras), se han transformado las rutas en variables binarias y se han depurado fugas de información. Sobre este escenario claramente desbalanceado, se han evaluado diversas estrategias de muestreo y clasificadores mediante validación cruzada anidada, utilizando el *F1-score* como métrica principal de selección y ajuste de hiperparámetros.

El flujo sin técnica de balanceo ha resultado el más robusto en cuanto a equilibrio entre estabilidad *inter-fold* y rendimiento en *test*, y, dentro de él, XGBoost el de mejor capacidad discriminativa. En alzhéimer, el modelo ha concentrado la mayor parte de los tratamientos reales en las primeras posiciones del *ranking*, mientras que, en esquizofrenia, la discriminación ha sido menor, lo que evidencia cómo la estructura interna de los datos condiciona el rendimiento final. La comparación con un *crank score*, basado en un enfoque estadístico lineal no supervisado, ha confirmado la superioridad de los métodos de *machine learning* para capturar relaciones complejas entre métricas heterogéneas.

Los resultados han demostrado que una capa integradora de *machine learning* puede unificar enfoques heterogéneos y generar *rankings* que orienten la validación experimental, reduciendo tiempo y costes frente al desarrollo *de novo*. No obstante, su fiabilidad depende de la calidad de los datos, por lo que deben interpretarse como guía preliminar. Por ello, las líneas futuras incluyen ampliar las fuentes de DRIVE, explorar búsquedas de hiperparámetros más amplias e incorporar técnicas para reforzar la confianza clínica. Solo la combinación de modelos *in silico* expresivos con evidencias experimentales sólidas permitirá que esta aceleración teórica se traduzca en beneficios terapéuticos tangibles sin comprometer la seguridad ni la eficacia clínica.

El trabajo incluye, además, un análisis de impacto personal, empresarial, social, económico, medioambiental y cultural, y alinea sus conclusiones con los Objetivos de Desarrollo Sostenible de la Agenda 2030, subrayando la necesidad de avanzar hacia una innovación farmacéutica responsable y sostenible.

Abstract

Traditional drug development requires millions of dollars in investment, more than a decade of work and involves high failure rates. Hence there is growing interest in drug repurposing, a strategy that seeks to assign new therapeutic indications to already approved drugs. However, the diversity of computational models (based on graphs, proximity in biomolecular networks or multiscale pathways) generates independent results that make it difficult to select candidates with a higher probability of clinical success. This Final Degree Project addresses this gap through the design and implementation of a machine learning metamodel, which integrates and synthesises information from three families of models stored in the DRIVE project database (including repositioning hypotheses generated by Graph Neural Networks, Network Proximity and Information Paths).

Following a thorough review of the state of the art, a unified dataset was constructed, comprising more than 100,000 disease-drug pairs and twelve predictor metrics. Pathways were transformed into binary variables and information leaks were cleaned. Given this clearly unbalanced scenario, several resampling strategies and classifiers have been evaluated by nested cross-validation, using the F1-score as the main metric for hyperparameter selection and adjustment.

The non-resampled workflow technique was the most robust in terms of balance between inter-fold stability and test performance, and within it, XGBoost was the one with the best discriminative capacity. In Alzheimer's disease, the model has concentrated most of the actual treatments in the top positions of the ranking. In contrast, in schizophrenia, the discrimination has been lower, which evidences how the internal structure of the data conditions the final performance. Furthermore, the comparison with a *crank score*, based on an unsupervised linear statistical approach, has confirmed the superiority of machine learning methods for capturing complex relationships between heterogeneous metrics.

Results demonstrated that an integrating machine learning layer can unify heterogeneous approaches and generate rankings to guide experimental validation, reducing time and cost compared to de novo drug development. However, their reliability depends on the quality of the data, so they should be interpreted as a preliminary guide. Future lines of work therefore include expanding DRIVE sources, exploring broader hyperparameter searches, and incorporating techniques to strengthen clinical confidence. Only the combination of expressive *in silico* models with robust experimental evidence will allow this theoretical acceleration to translate into tangible therapeutic benefits without compromising safety and clinical efficacy.

In addition, this final project also includes an analysis of personal, business, societal, economic, environmental and cultural impact, and aligns its conclusions with the Sustainable Development Goals of the 2030 Agenda, thereby underlining the need to move towards responsible and sustainable pharmaceutical innovation.

Tabla de contenidos

1	Introducción	1
1.1	Contexto	1
1.2	Objetivos	3
1.3	Plan de trabajo	4
1.4	Estructura del proyecto	5
2	Estado del arte	6
2.1	Modelos de reposicionamiento de fármacos	6
2.1.1	Modelos experimentales	6
2.1.2	Modelos computacionales	6
2.1.2.1	Modelos computacionales en DRIVE	8
2.2	Métodos de clasificación de <i>machine learning</i>	9
2.2.1	Métodos lineales	9
2.2.2	Métodos basados en árboles	10
2.2.3	Métodos probabilísticos	10
2.2.4	Métodos <i>lazy learning</i>	11
2.2.5	Métodos basados en redes	11
3	Desarrollo	12
3.1	Entorno de trabajo y herramientas utilizadas	12
3.1.1	Sistema operativo y entorno de desarrollo	12
3.1.2	Estructura del proyecto	12
3.1.3	Librerías y herramientas utilizadas	13
3.2	Metodología	14
3.2.1	Comprensión del negocio	15
3.2.2	Comprensión de los datos	15
3.2.2.1	Descripción de las tablas fuente	16
3.2.2.2	Ingesta, almacenamiento y procesamiento inicial	17
3.2.2.3	Técnicas de Análisis Exploratorio de los Datos (EDA)	18
3.2.3	Preparación de los datos	19
3.2.4	Modelado	21
3.2.4.1	Estrategias de <i>sampling</i>	22
3.2.4.2	Métodos de clasificación	24
3.2.4.3	Selección del modelo - Validación cruzada anidada	24
3.2.4.4	Búsqueda de hiperparámetros y ajuste de umbral	24
3.2.5	Evaluación	25

3.2.5.1 Métricas de evaluación	25
3.2.5.2 Comparación externa – <i>Crank Score</i>	26
3.2.6. Despliegue	27
4 Resultados y discusión	28
4.1 Resultados análisis exploratorio.....	28
4.2 Estadísticas de la tabla final	31
4.3 Evaluación de los Modelos	32
4.3.1 Comparación de los <i>pipelines</i>	33
4.3.2 Comparación de los métodos de clasificación en <i>No Sampling</i>	38
4.3.3 Comparación global entre alzhéimer y esquizofrenia	42
4.3.4 Resultados de las predicciones	42
4.4. Comparación con Crank	44
5 Conclusiones y líneas futuras	47
5.1 Conclusiones	47
5.2 Líneas Futuras	48
6 Análisis de Impacto	50
6.1. Impacto personal	50
6.2. Impacto empresarial	50
6.3. Impacto social.....	51
6.4. Impacto económico	51
6.5. Impacto medioambiental.....	52
6.6. Impacto cultural	52
6.7. Contribución a los Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030	53
7 Bibliografía	55
Anexos.....	60
Anexo I: Configuración de hiperparámetros	61
Anexo II: Distribución de las métricas.....	62
Anexo III: Resultados completos de la validación cruzada	64

Índice de Figuras

Figura 1. Diagrama de Sankey que representa la cantidad de fármacos que finalmente se utilizan para tratar enfermedades. Tomada de: [5].....	2
Figura 2. Comparación entre el enfoque tradicional (<i>de novo</i>) y el reposicionamiento de fármacos. Adaptada de: [5].....	2
Figura 3. Diagrama de Gantt que resume la planificación del presente TFG....	5
Figura 4. Tipos de métodos de clasificación de ML.	9
Figura 5. Estructura del proyecto.	13
Figura 6. Diagrama proceso CRISP-DM. Adaptado de: [42].....	14
Figura 7. Rutas de conexión multi-escala entre enfermedad-fármaco.....	17
Figura 8. Flujo de descarga, procesamiento y EDA de las tablas.	18
Figura 9. Flujo de modelado: técnicas de balanceo, validación cruzada anidada con búsqueda de hiperparámetros y evaluación de clasificadores.	22
Figura 10. Métodos de balanceo de la variable clase. Tomada de: [45] [46]....	23
Figura 11. Frecuencia de aparición de cada <i>Information Path</i>	28
Figura 12. <i>Heatmap</i> de coocurrencia entre <i>Information Paths</i>	29
Figura 13. <i>Boxplot</i> de la variable <i>path 2</i> según la clase.....	30
Figura 14. Matriz de correlación de Spearman entre las variables del <i>dataset</i> unificado.....	31
Figura 15. Comparación <i>inter-fold</i> de las métricas promedio para cada <i>pipeline</i> y métodos durante la validación cruzada.	34
Figura 16. Comparativa de clasificadores sin muestreo en función de las métricas clave de evaluación, desglosadas por enfermedad.	39
Figura 17. Evolución <i>Recall@K</i> de los métodos de clasificación <i>No Sampling</i> en alzhéimer y esquizofrenia.	40
Figura 18. Evolución <i>Precision@K</i> del de los métodos de clasificación <i>No Sampling</i> en alzhéimer y esquizofrenia.	40
Figura 19. Curva AUC-ROC de XGBoost en alzhéimer y esquizofrenia.	41
Figura 20. Matriz de Confusión de XGBoost en alzhéimer y esquizofrenia....	41
Figura 21. Distribución de pares con <i>crank score</i> > 0.9 por clase para alzhéimer y esquizofrenia.	46
Figura 22. Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030. Fuente: [64]	53

Índice de Tablas

Tabla 1. Modelos computacionales y sus tablas asociadas en la base de datos de DRIVE.	15
Tabla 2. Ejemplo de los datos antes de ser transformados.	20
Tabla 3. Ejemplo de los datos después de ser transformados.	20
Tabla 4. Ejemplo de los datos filtrando por class = 0 (<i>unknown</i>).	20
Tabla 5. Ejemplo de los datos filtrando por class = 1 (<i>treatment</i>).	20
Tabla 6. Ejemplo de la tabla final con los datos integrados.	21
Tabla 7. Matriz de Confusión.	26
Tabla 8. Estadísticas descriptivas de la tabla integrada.	32
Tabla 9. Resultados de los métodos en los cinco <i>pipelines</i> para alzhéimer.	36
Tabla 10. Resultados de los métodos en los cinco <i>pipelines</i> para esquizofrenia.	37
Tabla 11. Predicciones de XGBoost en alzhéimer.	43
Tabla 12. Predicciones XGBoost esquizofrenia.	43
Tabla 13. Comparación resultados XGBoost frente a <i>crank score</i> para alzhéimer.	45
Tabla 14. Comparación resultados XGBoost frente a <i>crank score</i> para esquizofrenia.	45

Acrónimos y Abreviaturas

AUC-ROC	<i>Area Under the ROC Curve</i>
CTB	<i>Centro de Tecnología Biomédica</i>
CRISP-DM	<i>Cross-Industry Standard Process for Data Mining</i>
DR	<i>Drug Repurposing</i>
DRIVE	<i>Drug Repurposing In Vivo Environment</i>
EDA	<i>Exploratory Data Analysis</i>
EHR	<i>Electronic Health Records</i>
FN	<i>False Negative</i>
FP	<i>False Positive</i>
GNN	<i>Graph Neural Network</i>
GWAS	<i>Genome-Wide Association Study / Studies</i>
IA	<i>Inteligencia Artificial</i>
KNN	<i>k-Nearest Neighbours</i>
MEDAL	<i>Medical Data Analytics Laboratory</i>
ML	<i>Machine Learning</i>
ROC	<i>Receiver Operating Characteristic</i>
SGBD	<i>Sistema Gestor de Bases de Datos</i>
SMOTE	<i>Synthetic Minority Over-sampling Technique</i>
SVM	<i>Support Vector Machine</i>
TFG	<i>Trabajo de Fin de Grado</i>
TN	<i>True Negative</i>
TP	<i>True Positive</i>
UPM	<i>Universidad Politécnica de Madrid</i>
XGBoost	<i>eXtreme Gradient Boosting</i>

1 Introducción

1.1 Contexto

El desarrollo tradicional de nuevos fármacos es un proceso largo, costoso y con una alta tasa de fracaso [1]. Este complejo procedimiento suele iniciarse con la investigación de las causas de la enfermedad y de posibles dianas terapéuticas, para identificar (tamizando entre un ingente número de moléculas) compuestos, llamados *hits*, que se unan a dichas dianas o bien tengan actividad biológica; de entre todos ellos se buscarán los compuestos líderes (generalmente generados por modificaciones químicas) con propiedades farmacológicas óptimas. Estos procedimientos suelen realizarse con técnicas experimentales, aunque pueden agilizarse con técnicas computacionales. Tras esta fase de ensayo *in vitro* se harán progresivos ensayos en líneas celulares, animales y en humanos. Tras superar todas estas etapas, el nuevo fármaco precisa que un organismo regulador apruebe su uso clínico [2]. Se estima que este proceso, partiendo desde cero, puede durar entre 10 y 17 años, requiere una inversión media de 2,600 millones de dólares, y presenta una tasa de éxito inferior al 2.01% [3].

Esta situación ha provocado la búsqueda de estrategias alternativas, entre las que destaca el reposicionamiento de fármacos, por sus siglas en inglés, *Drug Repurposing* (DR), una técnica que consiste en identificar nuevas indicaciones terapéuticas para medicamentos ya existentes y aprobados, aunque no necesariamente con las mismas dosis [4]. El reposicionamiento es especialmente relevante si consideramos la gran cantidad de posibles moléculas con potencial terapéutico y el bajo porcentaje que finalmente llega a convertirse en un fármaco disponible en el mercado. Sabemos que los fármacos, en su mayoría, se componen de moléculas, y que el número de estas moléculas orgánicas factibles de sintetizar llega a 10^{60} estructuras, aunque solo se han conseguido sintetizar alrededor de 100 millones de ellas a lo largo de la historia. De ese conjunto, solo unas 4,000 han sido aprobadas por organismos reguladores, de las cuales aproximadamente 1,500 son fármacos que pueden utilizarse para tratar unas 4,500 enfermedades, cuando el número de enfermedades catalogadas oscila entre 10,000 y 24,000. Estas asociaciones terapéuticas, además, suelen reflejar tratamientos paliativos que no siempre conducen a la curación definitiva, sino que alivian síntomas o ralentizan la progresión de la enfermedad [5]. Esto se refleja visualmente en la **Figura 1**, mediante un diagrama de Sankey que muestra la proporción de moléculas que acaban siendo utilizadas como tratamientos para enfermedades.

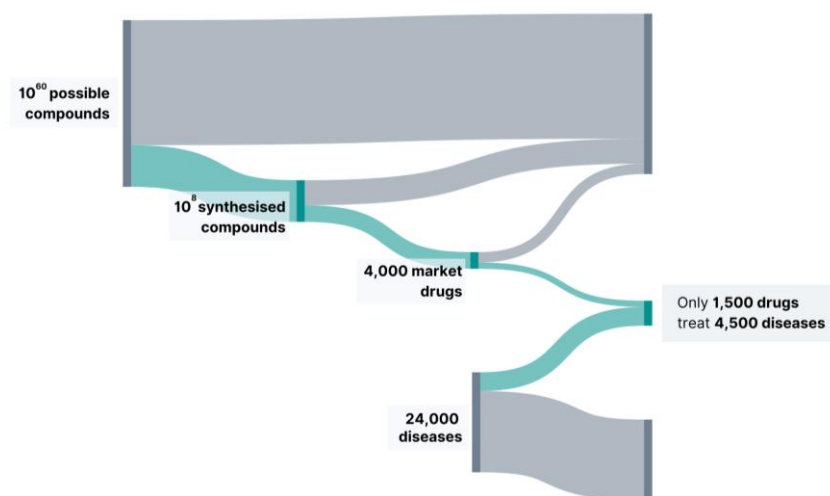


Figura 1. Diagrama de Sankey que representa la cantidad de fármacos que finalmente se utilizan para tratar enfermedades. Tomada de: [5].

En este contexto, el DR ofrece múltiples ventajas frente al enfoque tradicional. Esta estrategia aprovecha que estos fármacos ya han demostrado ser seguros en estudios preclínicos y en humanos, lo que reduce significativamente el riesgo de fracaso [6]. Además, permite acortar los tiempos de desarrollo (3–12 años frente a 10–17 años del enfoque tradicional) y disminuir la inversión necesaria [5]. Este enfoque ha resultado especialmente útil en contextos como la pandemia de COVID-19, donde se requería una respuesta terapéutica rápida; así como en enfermedades raras, para las cuales no siempre se destinan suficientes recursos debido a su baja rentabilidad comercial [6]. Esta diferencia de enfoques se muestra en la **Figura 2**, comparando las fases y los tiempos estimados de cada uno de ellos.

DESCUBRIMIENTO DE FÁRMACOS DE NOVO



REPOSICIONAMIENTO DE FÁRMACOS



Figura 2. Comparación entre el enfoque tradicional (*de novo*) y el reposicionamiento de fármacos. Adaptada de: [5].

No obstante, el DR también presenta obstáculos. Aunque el riesgo toxicológico es menor, hay barreras regulatorias, dificultades organizativas y problemas ligados a los derechos de propiedad intelectual. Estas limitaciones pueden frenar el avance de candidatos prometedores más allá de las fases iniciales de investigación [6].

El avance de las tecnologías de análisis molecular a gran escala, junto con el acceso a grandes volúmenes de datos biomédicos, ha facilitado el desarrollo de enfoques computacionales de DR. Los métodos basados en redes y técnicas de *Graph Deep Learning* permiten explorar la compleja red de interacciones entre genes, proteínas, síntomas, rutas biológicas y fármacos. Además, plataformas integradoras como DISNET¹, desarrollada en el laboratorio de Análisis de Datos Médicos (en inglés, *Medical Data Analytics Laboratory* (MEDAL)) del Centro de Tecnología Biomédica (CTB) de la Universidad Politécnica de Madrid (UPM), han reunido datos heterogéneos —biológicos, farmacológicos y fenotípicos— procedentes de fuentes públicas como PubMed, DisGeNET o DrugBank, estructurándolos en forma de redes complejas, lo que ha permitido desarrollar modelos para predecir asociaciones fármaco-enfermedad [7]. Esta plataforma ha permitido generar hipótesis de DR basadas en diferentes técnicas computacionales, almacenadas en una segunda base de datos llamada DRIVE², que sirve como base de datos central para este proyecto.

Sin embargo, cada uno de estos modelos se centra en aspectos específicos del problema; la proximidad en redes, la similitud fenotípica, el análisis de rutas biológicas o el uso de redes neuronales en grafos (*Graph Neural Networks* (GNNs), por sus siglas en inglés), lo que da lugar a una fragmentación que dificulta obtener una visión integrada y robusta de las hipótesis de reposicionamiento.

Por tanto, es necesario unificar estos enfoques mediante un metamodelo basado en *Machine Learning* (ML) que actúe como capa integradora. Así pues, este Trabajo de Fin de Grado (TFG) se centra en el desarrollo de dicho metamodelo, cuyo objetivo es combinar la información de diferentes modelos existentes, para mejorar la eficiencia en la identificación de nuevas indicaciones terapéuticas y proporcionar una herramienta robusta para la toma de decisiones en el contexto del DR.

1.2 Objetivos

El **objetivo principal** de este TFG es investigar y desarrollar un metamodelo basado en ML que permita integrar información procedente de distintos modelos existentes, con el fin de determinar cuál es la mejor hipótesis de reposicionamiento de fármacos entre los devueltos por los diversos modelos origen.

Para el desarrollo del proyecto, se han establecido los siguientes **objetivos secundarios**:

¹ <https://disnet.ctb.upm.es/>

² <https://live.drive-project.com/>

1. **Análisis de requisitos del proyecto en general.** Se llevará a cabo la revisión del estado del arte, es decir, se estudiarán diferentes modelos existentes de DR para comprender cómo funcionan, ya que serán las entradas del metamodelo. Además, será necesario definir el problema a resolver por el metamodelo: la falta de integración de las diferentes hipótesis de DR.
2. **Análisis detallado de los datos de origen.** Se accederá a las tablas de los diferentes modelos almacenados en la base de datos MySQL, donde se realizarán las transformaciones necesarias junto con un Análisis Exploratorio de los Datos (*Exploratory Data Analysis* (EDA), por sus siglas en inglés). Además, se utilizarán herramientas para manejar la gran cantidad de datos sin ocasionar problemas de memoria.
3. **Preparación del *dataset* para el entrenamiento del metamodelo.** Se desarrollará un *dataset* integrador a partir de la unificación de diversas tablas, consolidando las métricas relevantes para el DR. Además, una vez construido el *dataset*, se aplicarán técnicas de preprocesamiento adicionales en el caso de ser necesarias para optimizar el rendimiento del modelo.
4. **Planteamiento del metamodelo de *machine learning*.** Se diseñará la arquitectura del metamodelo. Para ello, será necesario decidir el algoritmo de ML a utilizar, cuáles serán los datos de entrada, métricas utilizadas y cómo medir su rendimiento.
5. **Creación del metamodelo.** Se implementará en Python la arquitectura establecida, se entrenará el modelo y se llevará a cabo el ajuste de hiperparámetros.
6. **Validación y pruebas.** Se realizará una evaluación de la eficacia del metamodelo y su capacidad predictiva.
7. **Documentación general del proyecto.** Se recopilará en la memoria toda la información del trabajo, es decir, justificación de la toma de decisiones, etapas de desarrollo, análisis de resultados, conclusiones, etc., y se llevará a cabo la presentación final.

1.3 Plan de trabajo

A continuación, se presenta en la **Figura 3** el diagrama de Gantt correspondiente a la planificación temporal del TFG. En él se detallan las diferentes fases del proyecto, desde la revisión del estado del arte hasta la elaboración final de la presentación. Cada una de estas tareas está distribuida a lo largo de las semanas de los meses de febrero a mayo, permitiendo visualizar de forma clara la duración estimada y el avance de estas. Además, observamos el porcentaje de duración de cada tarea respecto al total de horas.

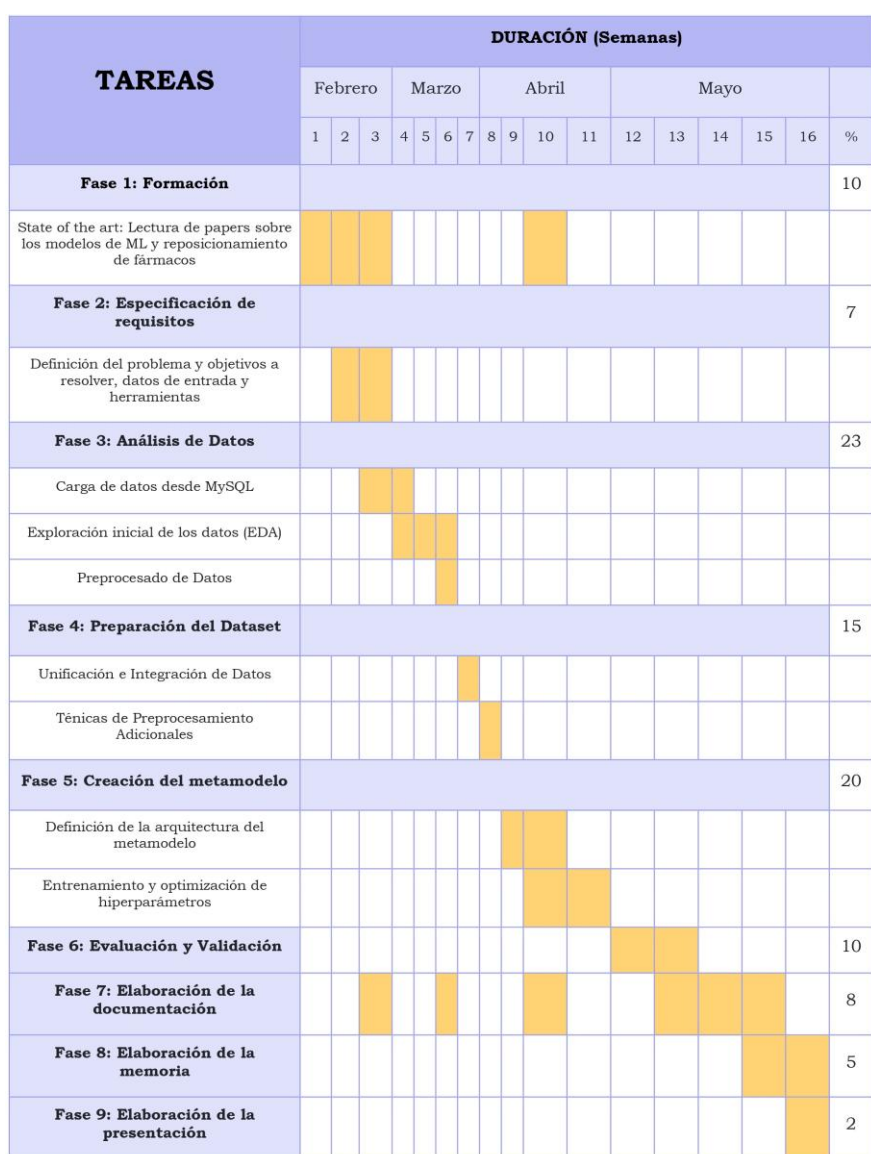


Figura 3. Diagrama de Gantt que resume la planificación del presente TFG.

1.4 Estructura del proyecto

A lo largo del proyecto se llevan a cabo todas las fases necesarias para la creación de un metamodelo de ML orientado al reposicionamiento de fármacos. En el **Capítulo 2** se analizan los principales modelos de DR, experimentales y computacionales, junto con los distintos algoritmos de clasificación que servirán como base para la construcción del metamodelo. El **Capítulo 3** detalla el proceso metodológico: análisis exploratorio de las tablas de DRIVE, integración de los distintos modelos en una única tabla, preprocesado de los datos y entrenamiento del metamodelo con varios clasificadores, evaluando su capacidad predictiva y robustez. En el **Capítulo 4** se presentan y discuten los resultados obtenidos; el **Capítulo 5** sintetiza las conclusiones y plantea líneas futuras de investigación; por último, el **Capítulo 6** reflexiona sobre el impacto personal, social, económico, medioambiental y cultural de la propuesta.

2 Estado del arte

2.1 Modelos de reposicionamiento de fármacos

El DR combina diversas estrategias para identificar nuevas indicaciones terapéuticas a partir de fármacos ya existentes. Este enfoque se apoya en la integración de amplios conjuntos de datos biomédicos y en el empleo de técnicas tanto experimentales como computacionales. Para la creación de este proyecto se utilizan diferentes modelos que, en conjunto, alimentan un metamodelo con el fin de priorizar las asociaciones fármaco-enfermedad. A continuación, se explican de manera general algunos de los principales enfoques existentes.

2.1.1 Modelos experimentales

Los enfoques experimentales intentan generar hipótesis de DR mediante la evaluación directa de compuestos en ensayos biológicos. Destacan dos estrategias principales, por un lado, los ***binding assays***, que buscan nuevas dianas terapéuticas a través de técnicas proteómicas. Por otro lado, el ***phenotypic screening***, que explora la actividad biológica de compuestos en modelos celulares u organismos sin necesidad de conocer la diana específica. Aunque estos métodos proporcionan evidencia sólida sobre la efectividad potencial de un fármaco, resultan laboriosos, costosos y difíciles de escalar, debido al conocimiento incompleto sobre los mecanismos moleculares que subyacen en muchas enfermedades [5].

2.1.2 Modelos computacionales

A diferencia de los enfoques experimentales, los modelos computacionales para el DR se basan en el análisis de grandes volúmenes de datos biomédicos, generalmente disponibles en bases de datos públicas. Estas estrategias permiten generar hipótesis de forma más rápida y con menor inversión, aunque pueden requerir información estructural sobre las proteínas diana y perfiles transcriptómicos o fenotípicos inducidos por los fármacos [5].

Entre los principales enfoques computacionales se encuentran los siguientes:

- ***Signature matching (comparación de firmas)***: Este enfoque compara la firma molecular de un fármaco (obtenida a partir de datos transcriptómicos, proteómicos y metabolómicos; estructuras químicas; o perfiles de eventos adversos) con la firma asociada a una enfermedad o a otro fármaco [8]. Si un fármaco es capaz de revertir el perfil de expresión génica característico de una enfermedad, se considera un posible candidato terapéutico [9]. Se diferencian tres modelos para obtener la firma del fármaco: firma génica, firma química y firma según eventos adversos. (i) **Firma génica**, donde el grado de correlación negativa entre la firma génica del fármaco y la de la enfermedad (es decir, los genes sobreexpresados en la enfermedad se encuentran reprimidos tras el tratamiento, y viceversa) permite inferir si el fármaco podría tener un efecto terapéutico sobre dicha enfermedad. Asume que si un fármaco puede revertir el patrón de expresión génica característico de un fenotipo

patológico y lo acerca al perfil sano, dicho fármaco podría revertir también el fenotipo de la enfermedad [6]. (ii) **Firma química**, basada en comparar la firma química de dos fármacos e identificar similitudes que podrían reflejar actividades biológicas compartidas. El proceso consiste en seleccionar un conjunto de características químicas de cada fármaco y construir redes basadas en los elementos comunes [6]. Presenta limitaciones como la posibilidad de que aparezcan metabolitos activos con efectos diferentes. (iii) **Firma según los eventos adversos**, que parte de la hipótesis de que dos fármacos que generan los mismos efectos secundarios podrían estar actuando sobre la misma diana, proteína o vía biológica; aunque en la práctica presenta muchas limitaciones [10].

- **Molecular docking:** Este enfoque permite predecir cómo interactúa un fármaco con su proteína objetivo, evaluando la complementariedad entre las estructuras tridimensionales de ambas moléculas. Puede aplicarse tanto para analizar una molécula frente a múltiples *targets* (*inverse docking*), como para evaluar diferentes compuestos sobre una misma diana terapéutica (*conventional docking*) [11].
- **Genome-wide association studies (GWAS):** Los GWAS tienen como objetivo identificar variantes genéticas asociadas a enfermedades comunes y, de este modo, aportar información sobre la biología de dichas enfermedades. Los datos obtenidos también pueden ayudar a identificar nuevas dianas terapéuticas, algunas de las cuales podrían estar compartidas entre enfermedades tratadas con fármacos y los fenotipos patológicos estudiados mediante GWAS, lo que podría abrir oportunidades para el DR. Este enfoque presenta limitaciones debido a la dificultad de interpretar y conocer el genoma humano de manera completa [12].
- **Pathway and network mapping:** Aunque algunos de los estudios anteriores pueden ser efectivos para identificar alteraciones genéticas que actúen directamente como diana terapéutica, en muchos casos estos genes no son óptimos desde el punto de vista terapéutico (*druggable*). Por esto, los enfoques basados en rutas moleculares (*pathway-based*) o en redes biomédicas (*network-based*) permiten construir mapas o redes que representan relaciones entre fármacos o enfermedades a partir de datos como patrones de expresión génica, fisiopatológica, interacciones proteicas o datos del genoma, con el objetivo de facilitar la identificación de candidatos para el reposicionamiento buscando genes próximos al identificado como diana terapéutica [13] [14].
- **Retrospective clinical analysis:** Este enfoque se basa en el análisis sistemático de grandes volúmenes de datos clínicos, provenientes de fuentes como las historias clínicas electrónicas, la farmacovigilancia post-comercialización o los resultados de ensayos clínicos. Las historias clínicas reúnen datos estructurados y no estructurados, lo que permite detectar, con alta potencia estadística, asociaciones entre fármacos y enfermedades. Sin embargo, aún existen desafíos importantes, como las

barreras éticas y legales que dificultan el acceso a los datos y la complejidad de procesar información no estructurada [6] [15].

- **Machine Learning:** Las técnicas de ML e Inteligencia Artificial (IA) son fundamentales para hallar nuevos usos a fármacos ya existentes, ya que permiten detectar de forma automatizada patrones en grandes volúmenes de datos moleculares, genéticos y clínicos. Con estas herramientas es posible predecir asociaciones fármaco-enfermedad, identificar nuevas dianas terapéuticas e incluso evaluar la seguridad de un fármaco en distintas indicaciones. Entre los métodos empleados destacan los modelos predictivos supervisados, como *Random Forest*, *Support Vector Machines* (SVM) y redes neuronales profundas [16] [17], y los enfoques basados en grafos que integran interacciones fármaco-diana con asociaciones gen-enfermedad [5] [18].

2.1.2.1 Modelos computacionales en DRIVE

Existen múltiples enfoques computacionales para DR, desde la comparación de firmas génicas hasta modelos de redes y GNNs. En este trabajo se han revisado seis familias de métodos, validadas en la literatura y almacenadas en tablas como modelos de DR en DRIVE, que sirven como punto de partida conceptual para el marco metodológico del TFG. Los detalles sobre la integración de algunas de ellas se describen posteriormente en el **Capítulo 3**.

- **GNNs:** Modelos de redes neuronales que operan sobre grafos, donde los nodos y aristas representan entidades biomédicas y sus relaciones. Al capturar simultáneamente la estructura local del grafo y las características individuales de los nodo, resultan especialmente adecuados para modelar interacciones complejas en DR y, con distintas arquitecturas y fuentes de datos, predecir enlaces enfermedad-fármaco que generan hipótesis de reposicionamiento [19] [20].
- **Análisis de conexiones en la red de conocimiento mediante una serie de Information Paths:** Propone cuatro rutas que parten de los síntomas y genes de una enfermedad para identificar posibles tratamientos, vinculando síntomas, enfermedades, genes, dianas y fármacos [21].
- **Proximidad de fármacos-enfermedades en el interactoma (Network Proximity):** Cuantifica las relaciones enfermedad-fármaco dentro del interactoma comparando la distancia observada entre proteínas asociadas con la distancia de pares generados aleatoriamente [22].
- **Basado en rutas biológicas (pathways):** Identifica un conjunto de casos, DREBIOP, por sus siglas en inglés (*Drug REpurposing based on BIOlogical Pathways*), que aprovechan el uso de rutas biológicas en lugar de dianas farmacológicas para reposicionar fármacos [23].
- **Basado en asociaciones entre enfermedades y rutas biológicas (disease pathways):** Identifica oportunidades de DR vinculando enfermedades a través de rutas biológicas compartidas, partiendo del

supuesto de que los fármacos para una enfermedad podrían tratar la otra [24].

- **Basado en valores umbral (*threshold values*) de asociación entre las enfermedades y diferentes entidades biológicas (genes, síntomas, fármacos):** Evalúa posibles casos de DR utilizando asociaciones enfermedad-gen determinadas por puntuaciones de DisGeNET previamente analizadas en escenarios de reposicionamiento conocidos [3].

2.2 Métodos de clasificación de *machine learning*

A continuación, se explican distintos tipos de métodos de clasificación que servirán como base para la construcción del metamodelo de reposicionamiento de fármacos de este proyecto. La **Figura 4** muestra una categorización general de los principales enfoques de ML empleados en tareas de clasificación.

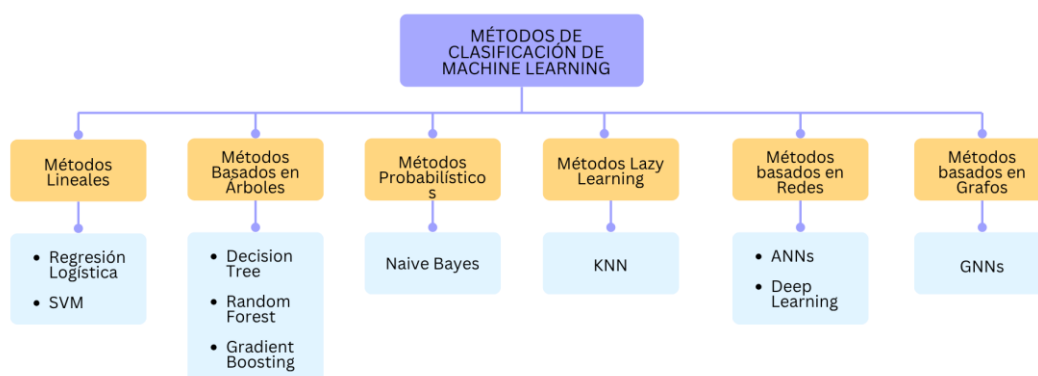


Figura 4. Tipos de métodos de clasificación de ML.

2.2.1 Métodos lineales

Los métodos lineales se caracterizan por asumir una relación lineal entre las variables de entrada (predictoras) y la salida del modelo. Esta dependencia lineal da lugar a una formulación bastante sencilla y a una buena interpretabilidad de cada variable. A pesar de su simplicidad, suelen proporcionar resultados sólidos. Entre los métodos más habituales se encuentran:

- ***Logistic Regression*:** Método supervisado utilizado principalmente en problemas de clasificación binaria. Calcula la probabilidad de pertenencia a una clase a partir de una combinación lineal de las variables predictoras, aplicando una transformación logística que garantiza que el resultado esté entre 0 y 1, permitiendo interpretar directamente el efecto de cada variable en dicha probabilidad. Sin embargo, presenta limitaciones cuando la relación entre las variables y la clase no es lineal o los datos presentan una alta complejidad [25].
- ***Support Vector Machines*:** Las SVM buscan el hiperplano que optimiza la separación entre clases en un espacio de alta dimensión, maximizando el margen entre los puntos de entrenamiento más cercanos y la frontera de decisión. Pueden utilizarse distintos *kernels* (lineal, polinómico, RBF,

etc.), para abordar problemas no lineales. Se caracterizan por su alto rendimiento en conjuntos de datos complejos o de tamaño moderado, y pueden dar buenos resultados incluso cuando la estructura de los datos no es estrictamente lineal. Sin embargo, es importante ajustar los hiperparámetros (parámetro C de regularización y los parámetros del *kernel*) para evitar problemas de sobreajuste³ [26].

2.2.2 Métodos basados en árboles

Los métodos basados en árboles se caracterizan por su enfoque jerárquico de división del espacio de características. A través de preguntas o reglas binarias, se crean nodos y hojas que dan como resultado una clasificación final de las instancias. Pueden manejar variables numéricas y categóricas, y cuya interpretabilidad es considerablemente buena en árboles poco profundos.

- **Decision Tree:** Se construyen dividiendo el espacio de características mediante reglas simples hasta lograr, en las hojas, una clasificación aceptable. Son muy intuitivos, pero tienden a sobreajustarse si no se aplican métodos adecuados de poda o regularización [27].
- **Random Forest:** Construye un conjunto (ensamble) de árboles de decisión, donde cada árbol se entrena sobre una muestra aleatoria del conjunto de entrenamiento y, en cada nodo, se selecciona aleatoriamente un subconjunto de características para decidir la división. La predicción final se obtiene mediante votación mayoritaria en clasificación. Esto mejora su capacidad de generalización, manteniendo bajo el riesgo de sobreajuste incluso al aumentar el número de árboles [28].
- **Gradient Boosting:** Construye secuencialmente árboles de forma aditiva, corrigiendo los errores de los árboles anteriores. De esta forma, se logran modelos potentes y flexibles, pero pueden ser propensos al sobreajuste si no se aplica el ajuste de hiperparámetros. En este contexto, destaca **Extreme Gradient Boosting (XGBoost)**, una implementación optimizada con técnicas de regularización, paralelización y eficiencia computacional, ampliamente utilizada por su rendimiento en ML [29].

2.2.3 Métodos probabilísticos

Naive Bayes: Se basa en un enfoque generativo, en el que se construye un método estadístico para estimar la probabilidad de cada clase dadas las características de entrada. Este método asume independencia condicional entre las variables predictoras (atributos). No obstante, suele funcionar bastante bien en muchos problemas de alta dimensionalidad, debido a su sencillez, velocidad y bajo requerimiento de datos [30].

³ El sobreajuste (*overfitting*) ocurre cuando el modelo se adapta en exceso a los patrones específicos —incluido el ruido— del conjunto de entrenamiento, perdiendo su capacidad para generalizar correctamente en nuevos datos.

2.2.4 Métodos *lazy learning*

K-Nearest Neighbors (KNN): Almacena las instancias del conjunto de entrenamiento y difiere la mayor parte del trabajo de cómputo hasta el momento de la predicción. Para clasificar un nuevo ejemplo, se buscan los k vecinos más cercanos según una métrica de distancia establecida, como la euclidiana o Manhattan, y se asigna la clase más frecuente entre ellos. Es un método bastante sencillo de entender e implementar, pero puede ser costoso si trabajamos con grandes volúmenes de datos, además de ser sensible a la elección de k y de la métrica de distancia. Este clasificador se engloba dentro de las técnicas de *lazy learning*, y sigue vigente hoy gracias a su simplicidad, flexibilidad y robustez en ciertos escenarios prácticos [31].

2.2.5 Métodos basados en redes

Dentro de los métodos discriminativos, las redes neuronales se han convertido en una herramienta muy importante para capturar relaciones complejas entre las variables de entrada y la variable objetivo, podemos distinguir:

- **Artificial Neural Networks:** Se inspiran en la estructura del sistema nervioso biológico. Están compuestas por capas de neuronas artificiales donde cada neurona combina las entradas ponderadas y las procesa a través de una función de activación. Este enfoque es muy flexible y puede aproximar funciones no lineales complejas, sin embargo, su entrenamiento puede ser bastante costoso si se aumenta el número de neuronas y capas [32].
- **Deep Neural Networks:** Están formadas por múltiples capas ocultas que permiten aprender representaciones jerárquicas de los datos, desde características simples hasta patrones complejos, lo que les otorga una gran capacidad de modelado. No obstante, se requieren grandes volúmenes de datos y una alta capacidad computacional para poder entrenar de forma eficiente las redes. Además, técnicas de regularización como la normalización o el *dropout* son fundamentales para evitar el sobreajuste, especialmente en redes muy profundas [33] [34].

Agregación de *rankings*

Un último enfoque relevante en DR se basa en la agregación de *rankings* generados por distintos modelos predictivos. Esta estrategia parte de la hipótesis de que distintos algoritmos (basados en IA, difusión en redes o medidas de proximidad) pueden capturar señales complementarias al evaluar las relaciones entre enfermedades y fármacos. Dado que los resultados producidos por cada técnica suelen presentar baja correlación entre sí, la combinación de sus *rankings* puede reforzar la robustez y precisión en la priorización de posibles candidatos. Para ello, se emplean métodos heurísticos de agregación inspirados en teorías de consenso, como *Borda count*, *Dowdall* o *CRank*, destacando este último por su eficacia al integrar el poder predictivo acumulado de modelos heterogéneos [35].

3 Desarrollo

3.1 Entorno de trabajo y herramientas utilizadas

3.1.1 Sistema operativo y entorno de desarrollo

Para el desarrollo del TFG se utilizó un equipo con sistema operativo Windows 11 y Python 3.11 como plataforma principal de programación. El IDE PyCharm 2022.2.2 se eligió por su capacidad de depuración de código y por la experiencia previa en su uso. Durante la elaboración del proyecto, se trabajó en un entorno local para las tareas de menor carga computacional. Sin embargo, debido a que el entrenamiento de los modelos resultó computacionalmente costoso, dicha parte del flujo se ejecutó en un servidor del laboratorio MEDAL, con sistema Ubuntu Server LTS 20.04.4 y GPU NVIDIA GeForce RTX 3090 (24GB). Además, al finalizar, el código fue publicado en GitLab, en un repositorio del laboratorio MEDAL⁴, para posible colaboración futura.

La fuente principal de información fue la base de datos DRIVE, alojada en un Sistema Gestor de Bases de Datos (SGBD) MySQL Server 8.0, de donde se extrajeron los datos de las tablas con las hipótesis de reposicionamiento generadas a partir de los distintos modelos computacionales, y que sirven como entrada del metamodelo.

3.1.2 Estructura del proyecto

El proyecto, como podemos observar en la **Figura 5**, se organizó siguiendo una estructura modular de carpetas que separa el código, los datos, los resultados y los gráficos. En la raíz del proyecto se encuentran los directorios principales de *code/*, *data/*, *notebooks/*, *results/* y *figures/*.

- **code/**: Contiene los scripts principales del proyecto: *models.py*, *metrics.py*, *preprocessing.py*, *run.py* y los *pipelines* de balanceo entre otros.
- **data/**: Se divide en subcarpetas *raw/*, *interim/* y *unified/* para gestionar las versiones de los datos en diferentes etapas del proceso.
- **notebooks/**: Incluye los notebooks de Jupyter utilizados durante el EDA, el preprocesado, la comparación con *Crank* y la generación de gráficas de los resultados.
- **results/**: Almacena las métricas, resultados generados por los modelos y *.pickles* de cada uno de ellos.
- **figures/**: Contiene las gráficas, visualizaciones y figuras empleadas en el análisis y documentación de resultados.

La documentación más detallada del código, scripts y datos utilizados se encuentra en los distintos archivos README.md del repositorio.

⁴ <https://medal.ctb.upm.es/internal/gitlab/grenada/drive-ml-metamodel>

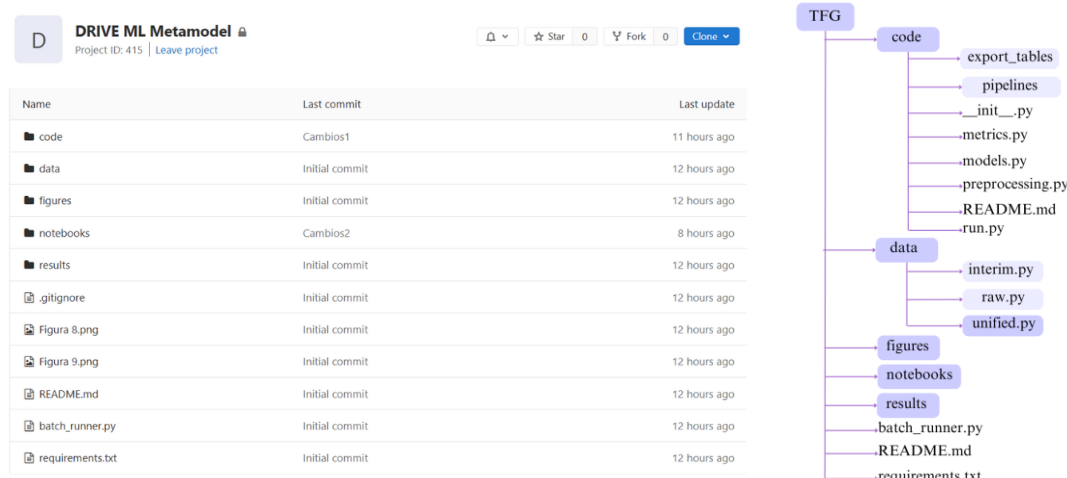


Figura 5. Estructura del proyecto.

3.1.3 Librerías y herramientas utilizadas

El flujo completo del proyecto, desde la extracción de datos, hasta el entrenamiento de los distintos modelos, se basó en un conjunto de librerías de Python que cubren almacenamiento eficiente, procesamiento distribuido, análisis exploratorio, aprendizaje automático y visualización. A continuación, se detallan las herramientas fundamentales utilizadas en el desarrollo del proyecto. La lista completa de bibliotecas y sus versiones correspondientes se encuentra disponible en el *requirements.txt* publicado en GitLab.

- **Pyarrow + Apache Parquet⁵ (versión 18.0.0):** Herramienta utilizada para optimizar el almacenamiento y transferencia de datos en formato columnar. Facilitó la lectura/escritura de las tablas en formato Parquet, reduciendo el tamaño en disco de los datos. Además, habilitó lectura por columnas durante las fases posteriores, algo muy importante ya que manejábamos millones de datos [36].
- **Dask⁶ (versión 2025.5.0):** *Framework* de computación paralela que extiende la funcionalidad de Pandas a entornos de mayor escala, mediante *dataframes* particionados y evaluación perezosa. Se utilizó para manejar las tablas de millones de filas que excedían la memoria disponible, permitiendo realizar transformaciones de forma diferida y computarlas solo cuando era necesario, facilitando el EDA sin desbordar la RAM, la exportación eficiente a Parquet y la paralelización de tareas básicas como *joins* o conversión de tipos [37].
- **Pandas⁷ (versión 2.2.3):** Librería fundamental para manipulación y análisis de datos estructurados en forma de *dataframes*. Se utilizó una vez materializados los datos (se redujo el número de filas), para realizar de forma ágil tareas como filtrado, combinación de tablas, cálculo de

⁵ <https://arrow.apache.org/docs/python/parquet.html>

⁶ <https://docs.dask.org/en/stable/>

⁷ https://pandas.pydata.org/docs/getting_started/overview.html

estadísticas descriptivas y generación de tablas resumen durante la fase de EDA [38].

- **Scikit-Learn⁸ (versión 1.6.1)**: Biblioteca principal de ML en Python, utilizada para implementar la mayoría de los algoritmos de clasificación y las técnicas de validación. En el proyecto fue el núcleo del modelado, debido a los clasificadores y herramientas de búsqueda de hiperparámetros que ofrece [39].
- **Matplotlib⁹ (versión 3.9.2) y Seaborn¹⁰ (versión 0.13.2)**: Bibliotecas de generación de gráficos utilizadas para la visualización de datos y resultados. Con matplotlib (a bajo nivel) y seaborn (a nivel más alto para gráficos estadísticos), se produjeron las diferentes visualizaciones que ayudaron a mostrar de forma gráfica los resultados más relevantes durante el análisis de datos [40].

3.2 Metodología

La metodología seleccionada para este proyecto se basa en **CRISP-DM (Cross-Industry Standard Process for Data Mining)** y se desarrolla a lo largo de las siguientes fases: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue. Cada una de estas etapas se adapta a las particularidades del reposicionamiento de fármacos mediante ML. A continuación, se ilustran estas fases en la **Figura 6** y se describen en detalle cada una de ellas [41].

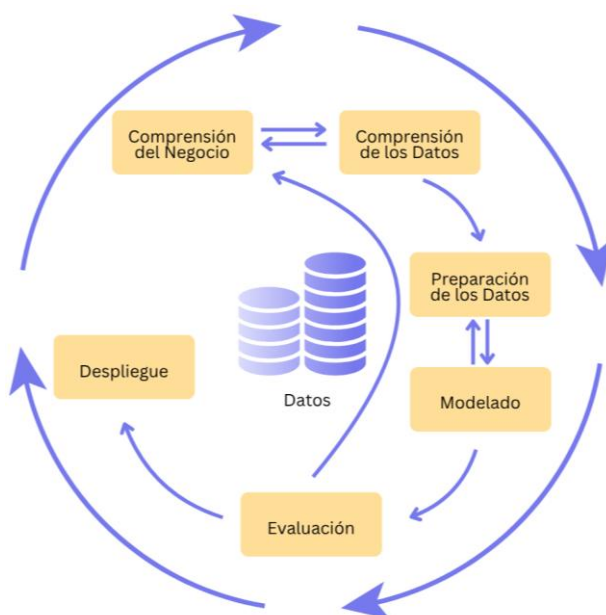


Figura 6. Diagrama proceso CRISP-DM. Adaptado de: [42].

⁸ https://scikit-learn.org/stable/user_guide.html

⁹ <https://matplotlib.org/stable/index.html>

¹⁰ <https://seaborn.pydata.org/>

3.2.1 Comprensión del negocio

El **reposicionamiento de fármacos** tiene como objetivo descubrir nuevas indicaciones terapéuticas para medicamentos ya aprobados, reduciendo el coste medio, el tiempo y los riesgos asociados al desarrollo *de novo*, tal y como se mencionó anteriormente. Plataformas como DRIVE recopilan modelos computacionales diversos (basados en redes, rutas biológicas o grafos), que generan hipótesis terapéuticas independientes entre sí.

Sin embargo, la falta de integración entre estos enfoques dificulta la priorización de candidatos con mayor probabilidad de éxito clínico. Por ello, en este proyecto se plantea el desarrollo de un metamodelo de ML capaz de combinar los resultados de distintos modelos de DR, asignando a cada par enfermedad-fármaco una puntuación integrada que facilite la toma de decisiones.

3.2.2 Comprensión de los datos

DRIVE almacena los resultados de los modelos de DR en tablas MySQL independientes. De todos los recursos enumerados en el **Capítulo 2**, este proyecto emplea únicamente tres de estas tablas como variables de entrada, junto con una tabla externa para la generación de la variable objetivo (*target*).

En la **Tabla 1** se muestra, de forma estructurada, el conjunto de modelos computacionales y las tablas de DRIVE que contienen sus resultados. El metamodelo desarrollado en este TFG integra únicamente *dr_gnns*, *dr_network_proximity* y *dr_information_paths*, seleccionadas por ser representativas de tres de los principales enfoques analizados: modelos basados en grafos, proximidad en redes biomoleculares y rutas multiescala.

Modelo Computacional	Tabla
Redes Neuronales en Grafos (GNN)	<i>dr_gnns</i>
Análisis de conexiones en la red de conocimiento mediante una serie de <i>Information Paths</i>	<i>dr_information_paths</i>
Proximidad de fármacos-enfermedades en el interactoma (<i>Network Proximity</i>)	<i>dr_network_proximity</i>
Basado en rutas biológicas (<i>pathways</i>)	<i>dr_pathways</i>
Basado en asociaciones entre enfermedades y rutas biológicas (<i>disease pathways</i>)	<i>dr_diseasepathways</i> y <i>dr_pathways_count</i>
Basado valores umbral (<i>threshold values</i>) de asociación entre las enfermedades y diferentes entidades biológicas (genes, síntomas, fármacos)	<i>dr_threshold_values</i>

Tabla 1. Modelos computacionales y sus tablas asociadas en la base de datos de DRIVE.

3.2.2.1 Descripción de las tablas fuente

- **Tabla dr_gnns**

Esta tabla contiene las predicciones generadas por el modelo BEHOR, el cual utiliza técnicas basadas en grafos para identificar posibles asociaciones terapéuticas entre enfermedades y fármacos. Así pues, cada fila representa un par enfermedad-fármaco identificado por los campos `disease_id` y `drug_id`, acompañado de múltiples métricas que reflejan distintas estrategias de predicción: **dmsr**, **dmsrC**, **behor** y **behorC**. Para todas estas métricas, valores más elevados (cerca de 1) representarán una mayor probabilidad de candidato a reposicionamiento para dicho par [19].

- **Tabla dr_network_proximity**

Esta tabla recoge las métricas generadas a partir del modelo de proximidad en redes biomoleculares. La hipótesis principal de dicho enfoque es que cuanto más cercanas estén las dianas de un fármaco al conjunto de genes causales de una enfermedad, mayor será su probabilidad de ser terapéuticamente eficaz [22].

En dicha tabla, al igual que antes, cada fila representa un par enfermedad-fármaco (`disease_id`, `drug_id`), junto con una serie de métricas que permiten interpretar esta proximidad: **closest_distance** (menor valor, más cercanía, mayor probabilidad de ser candidato), **dc_mean**, **dc_std** y **proximity** (cuanto más negativo sea el valor de `proximity`, mayor es la probabilidad de que el fármaco sea un buen candidato a reposicionamiento).

- **Tabla dr_information_paths**

Dicha tabla contiene asociaciones enfermedad-fármaco generadas a través de distintas rutas de conexión definidas sobre una red biomédica multi-escala. Estas rutas combinan información fenotípica, biológica y farmacológica, permitiendo identificar conexiones indirectas entre entidades clínicas [21]. En la tabla se incluyen únicamente los **path 1, 2, 4 y 5**, descartando *path 3*, por presentar resultados inconsistentes en evaluaciones previas.

Cada fila representa un par enfermedad-fármaco (`disease_id`, `drug_id`) identificado mediante una ruta específica (`path_id`). Un mismo par puede aparecer varias veces si ha sido detectado por más de una ruta. La tabla contiene únicamente tres variables: **disease_id**, **drug_id** y **path_id**.

Las diferentes rutas, incluida *path 3*, se describen a continuación en la **Figura 7**, indicando los nodos intermedios que conectan la enfermedad con el fármaco:

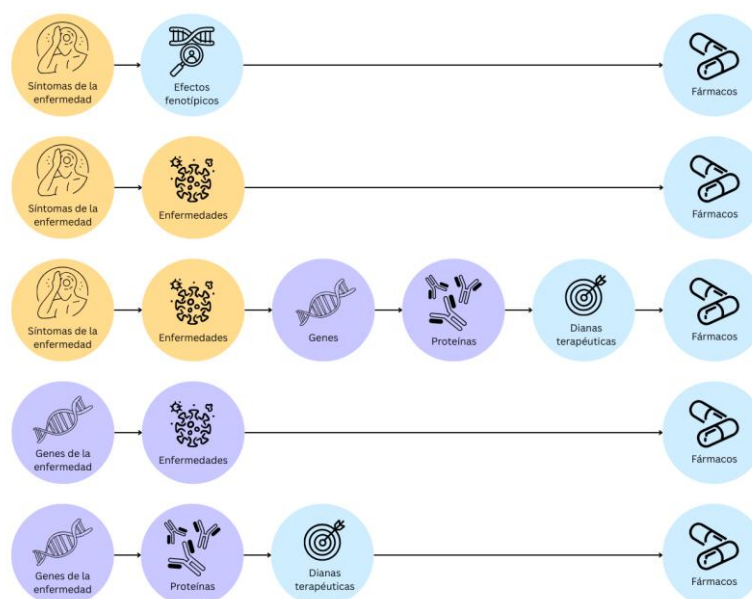


Figura 7. Rutas de conexión multi-escala entre enfermedad-fármaco.

Estas rutas permiten identificar asociaciones indirectas que podrían pasar desapercibidas en enfoques tradicionales de DR, al integrar distintas capas del conocimiento biomédico.

Variable objetivo (*target*)

Para entrenar el metamodelo como un problema de clasificación binaria, se ha definido una variable objetivo (*class*) que indica si una asociación enfermedad-fármaco ha sido validada clínicamente con anterioridad o no. Esta información se extrae de la tabla *disease_drug*, donde se establecerá como **1 (Treatment)** si el par aparece registrado como tratamiento aprobado, es decir, si se encuentra en dicha tabla, y como **0 (Unknown)** si no aparece en esta. No obstante, los detalles sobre la construcción final de esta variable se explicarán más adelante, en la **subsección 3.2.3 Preparación de los datos** del **Capítulo 3**.

3.2.2.2 Ingesta, almacenamiento y procesamiento inicial

Antes de llevar a cabo el EDA, fue necesario establecer un flujo de trabajo robusto de conexión, descarga, almacenamiento y manipulación de los datos, debido al gran volumen de información presente en las tablas utilizadas.

Conexión y descarga de datos: La mayoría de los datos se extrajeron directamente de la base de datos MySQL de DRIVE, mediante consultas SELECT utilizando *mysql-connector-python* desde Jupyter Notebooks. Para evitar errores de conexión por tiempo de espera y optimizar el uso de memoria, los datos se extrajeron en bloques (*chunks*) de 1 millón de filas. El flujo completo de conexión, descarga y análisis fue desarrollado íntegramente en Python.

No obstante, la tabla *dr_gnns* se descargó directamente desde un fichero CSV suministrado por el equipo de laboratorio MEDAL, convirtiéndolo a formato Parquet con particiones, al igual que el resto de tablas. Esto se debió a que, durante el EDA preliminar, se observó que todas las métricas presentaban

valores entre 0.5 y 0.7, señal de que los datos almacenados en MySQL eran incorrectos. El equipo confirmó que los datos eran inconsistentes y, aunque fueron corregidos, no fue posible cargarlos a tiempo en la base de datos debido a su gran volumen, por lo que se optó por facilitar el CSV original para continuar con el desarrollo del proyecto.

Almacenamiento eficiente: Para facilitar el procesamiento, los datos fueron almacenados en formato Parquet, lo que permitió reducir el uso de memoria mediante compresión y acceso por columnas. La tabla `dr_gnns` se fragmentó en 109 archivos Parquet de aproximadamente 1 millón de filas cada uno, `dr_network_proximity` generó un único archivo Parquet con 126,208 filas, y `dr_information_paths` se dividió en 16 archivos (15 con 1 millón de filas y 1 final con 68,726 filas) todas descargadas con el procedimiento mencionado anteriormente.

Procesamiento distribuido: Durante las etapas de unificación y limpieza se trabajó con *Dask DataFrame*, ya que permite operar sobre datos distribuidos sin cargarlos completamente en memoria. No obstante, tras el *inner join* final, al tener un menor número de filas, se materializó el *dataframe* con `.compute()` para trabajar directamente en Pandas y facilitar el análisis descriptivo.

Una vez descargados y almacenados los datos de manera eficiente, se procedió a su EDA con el fin de comprender la estructura y características de cada una de las tablas utilizadas. La **Figura 8** resume el flujo completo seguido desde la descarga hasta el análisis de las principales tablas de DRIVE.

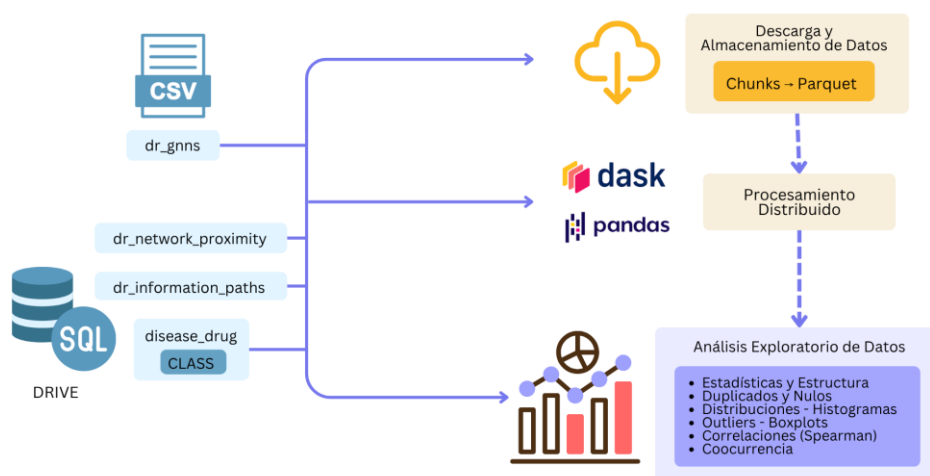


Figura 8. Flujo de descarga, procesamiento y EDA de las tablas.

3.2.2.3 Técnicas de Análisis Exploratorio de los Datos (EDA)

Para comprender la estructura en detalle de los tres modelos utilizados como entrada del metamodelo, `dr_gnns`, `dr_network_proximity` y `dr_information_paths`, junto a la tabla unificada, se llevó a cabo un análisis exploratorio de los datos. A continuación, se describen las técnicas utilizadas.

Inspección estructural inicial: Se aplicaron los métodos `.info()` y `.describe()` a cada una de las tablas para obtener una primera caracterización de las variables: tipo de datos, número de registros, valores

mínimos y máximos, media, desviación estándar, etc. También se obtuvieron estadísticas básicas como el número total de filas por tabla junto con el número de enfermedades y fármacos únicos.

Calidad de datos: Se evaluó la presencia de duplicados en los pares enfermedad-fármaco para asegurar la unicidad de los registros. Además, se cuantificó el número y el porcentaje de valores nulos por columna, diferenciando por clases (*treatment/unknown*) en algunos casos.

Distribución de variables y outliers: Se analizaron las distribuciones de todas las métricas utilizando histogramas (en escala logarítmica debido a la gran dispersión de valores) y *boxplots* para detectar posibles *outliers*, diferenciando en ambos casos según el valor de la variable objetivo (*class*).

Relaciones entre características: Se construyó la matriz de correlación para examinar relaciones entre variables numéricas, utilizando el **coeficiente de Spearman**.

Este mide la monotonía en la relación entre dos variables, sin necesidad de que esta relación sea lineal. Para calcularlo, primero se reemplazan los valores por sus rangos y luego se aplica la fórmula de Pearson (ρ_s) sobre esos rangos:

$$\rho_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

Donde d_i es la diferencia entre los rangos de cada observación y n el número total de observaciones. Se optó por utilizar Spearman porque es más robusto frente a distribuciones no normales, presencia de *outliers* y relaciones no lineales, condiciones que se cumplen en varias de las métricas analizadas

En base a los resultados obtenidos en la matriz de correlación, se seleccionaron pares de variables con alta, baja o nula correlación y se representaron mediante **diagramas de dispersión (scatter plots)**. Es importante mencionar que, al utilizar el coeficiente de Spearman, una correlación alta positiva indica que a medida que una variable aumenta, la otra también lo hace de forma monótona, aunque no necesariamente de manera lineal. Una correlación negativa sugiere una relación monótona decreciente, mientras que un valor cercano a cero implica ausencia de una relación monótona clara entre las variables.

Análisis de coocurrencia (Information Paths): Además del análisis de correlación, en un EDA preliminar, se estudió la coocurrencia entre rutas (*paths*) activadas en la tabla *dr_information_paths*. Se construyó una matriz de coocurrencia para identificar combinaciones frecuentes de rutas activas y se generó un gráfico de barras con la frecuencia de activación por ruta.

3.2.3 Preparación de los datos

Con el fin de obtener un conjunto de datos único, limpio y estructurado que sirviera de entrada del metamodelo, se aplicaron varias etapas de preprocesado.

Transformación de *dr_information_paths* (pivotado)

La tabla *dr_information_paths* fue transformada para representar cada par enfermedad-fármaco en una única fila, y varias columnas binarias (*path_1*,

path_2, path_4, path_5), donde un 1 indica que la ruta correspondiente conecta dicho par. Esta transformación permitió integrar fácilmente la tabla con el resto de fuentes de información. Para optimizar el procesamiento, se utilizó Dask, *pivot_table()*, y se verificó la consistencia de los resultados con la base de datos original a través de WorkBench. En la **Tabla 2** y la **Tabla 3** se muestran las primeras 5 filas antes y después de la transformación, respectivamente.

disease_id	drug_id	path_id
C4280456	CHEMBL1536	4
C4280456	CHEMBL1536	5
C4280456	CHEMBL477772	4
C4280456	CHEMBL477772	5
C4280456	CHEMBL723	4

Tabla 2. Ejemplo de los datos antes de ser transformados.

disease_id	drug_id	path_1	path_2	path_4	path_5
C4280456	CHEMBL1536	0	0	1	1
C4280456	CHEMBL477772	0	0	1	1
C4280456	CHEMBL723	0	0	1	1
C4280457	CHEMBL1200666	0	0	1	1
C4280457	CHEMBL2356023	0	0	1	1

Tabla 3. Ejemplo de los datos después de ser transformados.

Unión de dr_gnns con disease_drug (left join)

Para generar la variable objetivo (*class*), se realizó una unión *left join* tomando como base la tabla dr_gnns y cruzándola con disease_drug. De este modo, a cada par enfermedad-fármaco se le asignó un valor de 1 si también estaba presente en disease_drug (es decir, si era un tratamiento validado clínicamente), y 0 en caso contrario. En la **Tabla 4** y la **Tabla 5** se muestran las primeras 5 filas tras la creación de la variable objetivo, filtrando por cada uno de sus posibles valores.

disease_id	drug_id	behor	behorC	dmsr	dmsrC	class
C0000727	CHEMBL1000	0.055649213	0.11300971	0.06808827	0.28477213	0
C0000727	CHEMBL100116	0.1063099	0.14206292	0.0954956	0.2810142	0
C0000727	CHEMBL1002	0.033165563	0.044082023	0.027331356	0.15744208	0
C0000727	CHEMBL1003	0.10198709	0.21274272	0.093587846	0.49977058	0
C0000727	CHEMBL1004	0.033165563	0.03824563	0.027331298	0.10003094	0

Tabla 4. Ejemplo de los datos filtrando por class = 0 (unknown).

disease_id	drug_id	behor	behorC	dmsr	dmsrC	class
C0000727	CHEMBL370805	0.053151254	0.17756176	0.06674457	0.32185856	1
C0000729	CHEMBL1123	0.046619616	0.061136954	0.042296126	0.21691631	1
C0000729	CHEMBL469	0.13878408	0.19215986	0.10876556	0.6083914	1
C0000729	CHEMBL607	0.1126074	0.17551444	0.10052526	0.63307965	1
C0000729	CHEMBL80	0.6654998	0.95372856	0.69651306	0.99853027	1

Tabla 5. Ejemplo de los datos filtrando por class = 1 (treatment).

Integración final (*inner joins*)

Finalmente, se llevó a cabo la integración total de los datos de los tres modelos, junto con la variable objetivo (*class*). Para ello, se utilizaron los datos transformados junto con la tabla *Network_Proximity*, la cual habíamos descargado anteriormente directamente desde MySQL.

La integración se llevó a cabo mediante uniones internas (*inner join*), conservando únicamente aquellos pares enfermedad-fármaco presentes en las tres tablas. El conjunto resultante, más reducido, nos permitió aplicar `.compute()` y trabajar con la librería Pandas directamente en el procesamiento de los datos.

Finalmente, se eliminaron las filas con valores faltantes en algunas variables, pero no comprometió significativamente el tamaño ni la representatividad de la muestra, ya que constituían entre un 1% y un 3% de los datos. Además, se eliminó la variable *path 2*, como se detalla en el **Capítulo 4**. Tras este preprocesado, se obtuvo un conjunto de datos limpio, consistente y listo para ser utilizado en el entrenamiento del metamodelo, cuya estructura puede observarse en la **Tabla 6** para las primeras 5 filas.

disease_id	drug_id	Path 1	Path 4	Path 5	Closest distance	Dc_mean	dc std	proximity	behor	behorC	dmsr	dmsrC	class
C0001973	CHEMBL1000	0	1	0	2.00000	1.82700	0.391243	0.442181	0.999997	1.0	1.0	1.000000	0
C0001973	CHEMBL1005	0	1	1	1.33333	1.65933	0.255063	-1.278110	1.000000	1.0	1.0	1.000000	0
C0001973	CHEMBL1006	0	1	0	2.00000	1.90600	0.370357	0.253809	1.000000	1.0	1.0	1.000000	0
C0001973	CHEMBL1008	0	1	0	1.80000	1.88880	0.131351	-0.676051	0.999961	1.0	1.0	0.999993	0
C0001973	CHEMBL1009	0	1	1	1.20000	1.97955	0.209331	-3.724010	1.000000	1.0	1.0	1.000000	0

Tabla 6. Ejemplo de la tabla final con los datos integrados.

3.2.4 Modelado

Con el conjunto de datos ya integrado y preprocesado, la fase de modelado se centró en entrenar y comparar distintos clasificadores que actuaran como metamodelo, es decir, como capa integradora de las métricas procedentes de los tres modelos de DR de origen. Para ilustrar el flujo de trabajo llevado a cabo durante esta fase, la **Figura 9** muestra un esquema con las principales técnicas empleadas.

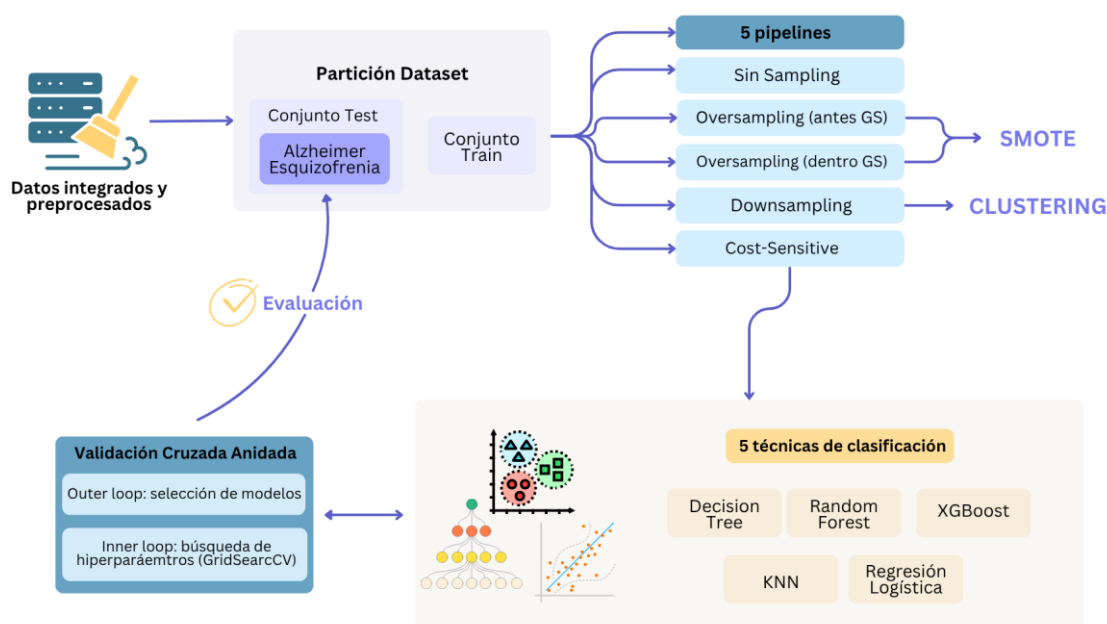


Figura 9. Flujo de modelado: técnicas de balanceo, validación cruzada anidada con búsqueda de hiperparámetros y evaluación de clasificadores.

Antes de iniciar el entrenamiento, se particionó el conjunto de datos. En concreto, se reservaron para el conjunto de test todos los pares enfermedad-fármaco correspondientes a dos patologías neurológicas de alto impacto social: la **enfermedad de Alzheimer** y la **esquizofrenia**. Estas dos enfermedades no solo presentan una elevada carga sociosanitaria, sino que además son representativas de áreas donde existe una necesidad urgente de nuevas opciones terapéuticas.

Ambas enfermedades fueron excluidas completamente del conjunto de entrenamiento, garantizando que los modelos no tuvieran acceso previo a información directa sobre estos casos. Durante la evaluación final, se aplicó el modelo por separado sobre cada una de ellas, permitiendo analizar su capacidad de generalización a nuevos contextos clínicos. El resto de las instancias disponibles se utilizaron para entrenamiento, aplicando validación cruzada anidada estratificada, para evaluar el rendimiento de los modelos y ajustar sus hiperparámetros sin necesidad de reservar explícitamente un subconjunto de validación adicional.

Además, dos de los algoritmos, KNN y Regresión Logística, son sensibles a la escala; por ello se aplicó estandarización (*z-score*) a sus datos de entrada. El resto de métodos sí son capaces de manejar valores no normalizados, por lo que recibieron los datos originales.

3.2.4.1 Estrategias de *sampling*

El conjunto final, tras la integración de los diferentes modelos de DR, estaba fuertemente desbalanceado, por lo que se construyeron cinco *pipelines* paralelos.

No sampling: El primer *pipeline* se lleva a cabo sin muestreo de los datos. Se toma el conjunto directamente de la etapa de preprocesado y se entrena en los diferentes modelos.

Oversampling: En este *pipeline*, se implementaron dos variantes complementarias basadas en técnicas de sobremuestreo sintético de la clase minoritaria (Synthetic Minority Oversampling Technique, **SMOTE**), tal y como se presenta en la **Figura 10.a**. Para cada registro etiquetado como ‘*treatment/1*’ se buscan sus vecinos más cercanos y se generan nuevos puntos artificiales a lo largo de los segmentos que los unen [43].

- En la primera variante, se aplica **SMOTE antes de la validación cruzada**, generando instancias sintéticas únicamente en el conjunto de entrenamiento.
- En la segunda variante, **SMOTE se incluye dentro de la validación cruzada**, aplicándose en cada *fold* durante el bucle interno de la validación cruzada. Esto evita fugas de información, ya que garantiza que las instancias sintéticas generadas a partir de los datos de entrenamiento no influyan en los subconjuntos usados como validación dentro de la *cross-validation*. Al evitar que los datos sintéticos contaminen el conjunto de validación, se obtienen estimaciones más realistas del rendimiento.

Downsampling: Existen diversas estrategias de *Downsampling* que permiten reducir la clase mayoritaria sin generar datos sintéticos, como se muestra en la **Figura 10.b**. Entre éstas se encuentran el submuestreo aleatorio, el método *NearMiss*, los vecinos más cercanos condensados, *Tomek Links* o el submuestreo basado en vecinos más cercanos editados [44].

En este proyecto, se optó por un enfoque guiado por *clustering*. Se aísla la clase mayoritaria (*unknown*, 0), se ejecuta un algoritmo de *K-means* y se fija el número de *clusters* igual al número de instancias de la clase minoritaria (*treatment*, 1). De cada clúster se conserva únicamente el elemento más cercano al centroide, ya que se busca reducir el tamaño de la Clase 0, sin generar muestras sintéticas y trabajar únicamente con los datos reales.

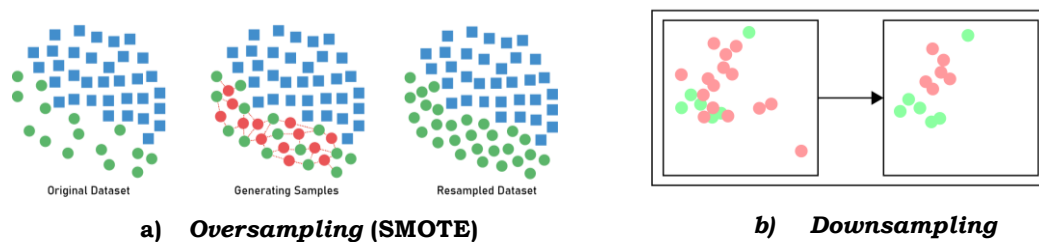


Figura 10. Métodos de balanceo de la variable clase. Tomada de: [45] [46].

Cost Sensitive: Por último, algunos algoritmos permiten corregir el desbalanceo directamente en la función de coste. Este enfoque, *Cost Sensitive*, asigna a la clase minoritaria (*treatment*, 1) un peso mayor, proporcionalmente inverso a su frecuencia, de modo que el modelo penaliza más los errores cometidos sobre ella, reduciendo así los falsos negativos. En modelos como *Decision Tree*, *Random Forest* y Regresión Logística se establece a través de `class_weight='balanced'`, mientras que en XGBoost con `scale_pos_weight=n_class_0 / n_class_1`.

3.2.4.2 Métodos de clasificación

Se seleccionaron cinco métodos de clasificación complementarios. **K-Nearest Neighbors** aporta una línea base muy intuitiva y dependiente de la geometría de los datos; **Regresión Logística**, ofrece una frontera de decisión interpretable y suele ser eficaz cuando existe una separación lineal entre clases; **Decision Tree** proporciona interpretabilidad y sirve de referencia para métodos de ensamble; **Random Forest** presenta aleatoriedad y votación mayoritaria para reducir varianza; y finalmente, **XGBoost**, un *gradient boosting* con regularización, permite capturar interacciones complejas y manejar desequilibrios. Con este conjunto se cubren modelos *lazy*, lineales, basados en árboles simples y ensambles aditivos, lo que permite comparar distintas vías de aprendizaje.

3.2.4.3 Selección del modelo - Validación cruzada anidada

Para estimar el rendimiento de cada clasificador, se aplicó **validación cruzada anidada estratificada**, utilizando 5 divisiones (*folds*) en el bucle externo (*outer loop*), de tal forma que, en cada iteración, uno de los subconjuntos actúa como validación externa y los cuatro restantes alimentan un *GridSearchCV* interno (*inner loop*), estratificado y con 5 *folds*, que busca la mejor combinación de hiperparámetros. Así, cada modelo se entrena únicamente con datos no vistos y se evita fuga de información (*data leakage*). Además, las métricas del pliegue externo se promedian al final de las cinco iteraciones, ofreciendo una estimación no sesgada aún en presencia de desbalanceo [47].

Esta técnica asegura que cada *fold* mantenga la proporción original de clases (tratamientos reales frente a pares desconocidos), lo cual resulta especialmente importante debido al desbalance del *dataset*.

3.2.4.4 Búsqueda de hiperparámetros y ajuste de umbral

La configuración óptima de cada clasificador se determinó mediante una búsqueda exhaustiva de hiperparámetros en cuadrícula (*GridSearchCV*), alojada en el bucle interno de la validación cruzada anidada descrita anteriormente. La métrica de referencia fue el **F1-score** por su capacidad para equilibrar precisión y *recall* en datos desbalanceados.

Para **KNN** se exploraron los valores de k (número de vecinos), el tipo de métrica de distancia (*euclidean* o *manhattan*) y la estrategia de ponderación; en **Random Forest** se variaron el número de árboles, la profundidad y el tamaño mínimo de nodo; en **Decision Tree** se ajustaron los mismos parámetros, añadiendo además diferentes funciones de impureza (*gini*, *entropy* y *log loss*); **Regresión Logística** combinó diferentes penalizaciones C , funciones de coste ($L1$ y $L2$) y solvers (*liblinear* y *saga*); **XGBoost** examinó el número de estimadores, profundidad de cada árbol y la tasa de aprendizaje (*learning rate*). La configuración completa de hiperparámetros explorada para cada modelo se detalla en el **Anexo I**.

Tras seleccionar los mejores hiperparámetros en cada iteración interna, el modelo resultante se evaluó en el pliegue externo correspondiente. Concluido el bucle externo, el modelo final se seleccionó en base al pliegue donde se obtuvo

un mayor *F1-score*. Las combinaciones óptimas por modelo y *pipeline* están disponibles en GitLab.

Además, se realizó el ajuste del **umbral de clasificación** (*threshold*) sobre los datos de validación interna. Por defecto, los clasificadores binarios consideran 0.5 como punto de corte para convertir probabilidades en predicciones (es decir, asignan la clase positiva si la probabilidad es ≥ 0.5). Sin embargo, debido al desbalance del *dataset*, se calculó el umbral óptimo como aquel que maximiza el *F1-score* sobre las predicciones del conjunto de validación.

Este umbral optimizado, junto con la mejor combinación de hiperparámetros, se aplicó después al conjunto de *test* reservado para las enfermedades seleccionadas (alzhéimer y esquizofrenia) para obtener las predicciones finales y las métricas de evaluación.

3.2.5. Evaluación

3.2.5.1 Métricas de evaluación

La calidad del metamodelo se midió siguiendo un esquema de métricas centrado en el ***F1-score***, tanto durante la evaluación interna de la validación cruzada anidada como en la evaluación final sobre los conjuntos de *test* por enfermedad.

Además, para cada métrica se calculó la **media $\pm$ desviación estándar** entre pliegues, lo que proporciona una estimación robusta del rendimiento esperado y permite identificar posibles signos de varianza excesiva o sobreajuste.

F1-score: Esta métrica es muy relevante en contextos biomédicos desbalanceados, donde los falsos negativos pueden suponer la pérdida de una oportunidad terapéutica, es decir, de un posible candidato a DR. Combina precisión y *recall* mediante:

$$F1 = 2 * \frac{(\text{Precisión} * \text{Recall})}{(\text{Precisión} + \text{Recall})}$$

Donde:

$$\text{Precisión} = \frac{TP}{TP + FP} \quad \text{Recall} = \frac{TP}{TP + FN}$$

Siendo, por sus siglas en inglés, TP los verdaderos positivos, FP los falsos positivos y FN los falsos negativos.

Además, ya que nos encontramos en un dominio biomédico altamente desbalanceado, conviene evaluar otras métricas:

- **Area Under the ROC Curve (AUC-ROC)**. Al disponer de predicciones probabilísticas, la curva ROC y su área permiten comprobar la discriminación global bajo distintos umbrales de decisión. Cuanto mayor es el AUC, mejor es la discriminación.
- **Precisión y recall**. En problemas de DR, es importante evaluar ambas métricas. La precisión es importante si se busca minimizar falsos positivos, es decir, no recomendar candidatos irrelevantes; mientras que,

si el riesgo está en descartar un tratamiento potencial a reposicionamiento, el *recall* tiene mayor importancia.

Además, se representan de forma visual ***recall@K*** y ***precision@K*** con el fin de acercarnos a un escenario real. El *recall@K* cuantifica el número de tratamientos verdaderos que aparecen entre los *K* primeros candidatos del *ranking*, es decir, estima cuántos candidatos a reposicionamiento serían detectados si se revisa solo ese tramo inicial. La *precision@K*, indica qué proporción de esos *K* pares priorizados resulta ser realmente positivos.

- ***Recall@100***. Para cada clasificador se ordenan los pares enfermedad-fármaco por probabilidad descendente y se cuentan cuántos tratamientos verdaderos (Clase 1) aparecen entre las 100 primeras filas. Esta métrica simula un escenario real de cribado, donde el investigador sólo explora un número limitado de hipótesis.
- **Matriz de confusión**. Permite visualizar de forma clara el número de verdaderos y falsos positivos/negativos, ayudando en la interpretación clínica de los resultados (**Tabla 7**).

	Predicción Class 0	Predicción Class 1
Valor Real Class 0	TN	FP
Valor Real Class 1	FN	TP

Tabla 7. Matriz de Confusión.

Siendo, por sus siglas en inglés, TP los verdaderos positivos, FP los falsos positivos, FN los falsos negativos y TN los verdaderos negativos. En el contexto de este trabajo, los FP presentan gran interés, ya que entre ellos pueden encontrarse nuevas hipótesis de reposicionamiento que el modelo considera prometedoras aunque aún no estén validadas. Por ello, en futuros estudios experimentales, sería recomendable priorizar la revisión de los primeros FP del *ranking* como posibles candidatos a testar.

3.2.5.2 Comparación externa – Crank Score

Por último, para disponer de un punto de referencia externo frente al que contrastar las probabilidades generadas por los clasificadores, se construyó un indicador agregado, *crank score*, a partir de las métricas que alimentan el metamodelo [48].

El procedimiento siguió una lógica de *ranking* por consenso inspirada en *Crank*:

- **Homogeneización del sentido de las métricas:** Para las cuatro métricas de *Network Proximity* (*closest_distance*, *dc_mean*, *dc_std* y *proximity*), un menor valor indica mayor probabilidad de eficacia como

candidato a reposicionamiento. Por ello, se invierten sus valores ($-x$) para que en todas las métricas un valor mayor indicara mejor candidato.

- **Ranking individual:** Cada par enfermedad-fármaco fue ordenado dentro de cada métrica, asignando posiciones más altas (M) a los pares más prometedores.
- **Agregación y normalización:** Los rankings fueron sumados para obtener una métrica continua, *crank_score*, posteriormente normalizada al rango $[0, 1]$ para facilitar su comparación directa con las probabilidades generadas por los modelos de clasificación.

Esta comparación externa permitirá valorar en qué medida el metamodelo replica, complementa o supera el ranking generado por la agregación directa de las métricas, aportando evidencia adicional sobre la validez práctica de las predicciones.

3.2.6. Despliegue

En este TFG, al ser un proyecto de investigación académica, no se ha abordado aún un despliegue operativo del metamodelo en forma de API ni su integración en entornos clínicos. No obstante, se han cuidado varios detalles para que el trabajo pueda reutilizarse y seguir ampliándose. Todo el proyecto se ha estructurado en módulos bien definidos, acompañados de varios ficheros README.md detallados que permiten su correcta reproducción. Además, se ha publicado en un repositorio privado de GitLab del laboratorio MEDAL, lo que facilita que cualquier miembro del equipo (o futuro colaborador) pueda reproducir los experimentos y, si lo desea, ensayar nuevas variantes del metamodelo.

La idea futura es integrar el metamodelo en las visualizaciones interactivas ya disponibles en la plataforma DRIVE. De este modo, los *rankings* generados podrán mostrarse junto a los resultados de los modelos individuales, permitiendo a los investigadores comparar predicciones, explorar de forma dinámica los pares enfermedad-fármaco priorizados según distintos criterios y obtener directamente los candidatos más prometedores para su validación experimental. Esta integración transformaría el metamodelo en una herramienta práctica dentro de DRIVE, reforzando la transferencia de conocimiento y facilitando la colaboración entre equipos clínicos y computacionales.

4 Resultados y discusión

4.1 Resultados análisis exploratorio

Durante la fase exploratoria, se llevaron a cabo tres análisis de los datos, dos de ellos preliminares y un último sobre la tabla integrada, con el fin de garantizar la solidez de los datos antes de entrenar el metamodelo y tomar decisiones de manera robusta. A continuación, se muestran los resultados que afectaron a la interpretación del conjunto de datos.

- **dr_gnns y disease_drug**

El primer análisis se realizó sobre la unión de las métricas de GNN y la variable *class* (*target*), donde se verificó la ausencia de duplicados en las columnas clave (*disease_id*, *drug_id*) y de valores nulos. El punto crítico fue el **desequilibrio de la Clase 1**, ya que había únicamente un 0.0554% de tratamientos validados frente a un 99.9446% de pares no confirmados (sobre un total de 102,354,688 instancias).

Destacó que las métricas de GNN presentaron valores mayores cuando los pares pertenecían a la Clase 1 (*treatment*), lo que podría ayudar a comprobar que dicho modelo base, entrada del metamodelo, funciona correctamente.

- **Information Paths**

El segundo análisis se centró en la tabla transformada de *Information Paths*. Tras verificar la integridad de los datos (sin duplicados ni nulos), se observó un **desequilibrio notable en la las vías de información**; *path* 4 aparecía en un 97.65% de los pares, mientras que *path* 5, *path* 1 y *path* 2 únicamente lo hacían en un 2.78%, 1.65% y 0.31% respectivamente. Este desequilibrio se evidencia de manera clara en el siguiente gráfico de barras de la **Figura 11**.

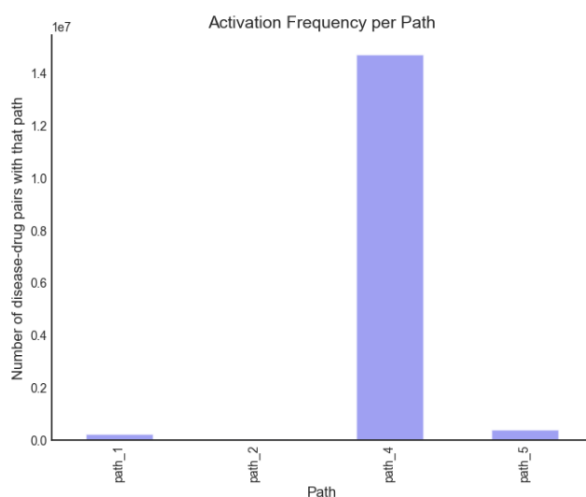


Figura 11. Frecuencia de aparición de cada Information Path.

Adicionalmente, se realizó un **análisis de coocurrencia**, es decir, se exploró la frecuencia con la que las vías aparecen simultáneamente. Se corroboró la elevada frecuencia de aparición del *path* 4, apareciendo en más de 14 millones

de pares. Las combinaciones duales fueron poco frecuentes, aunque destacó la aparición conjunta de *path* 4 y *path* 5, en 244,506 casos. Sin embargo, la aparición simultánea de las cuatro rutas fue poco destacable, con únicamente 245 pares. Esto se puede visualizar en la **Figura 12**.

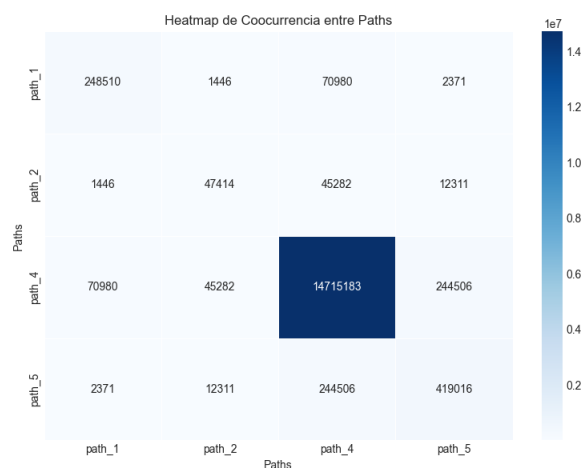


Figura 12. Heatmap de coocurrencia entre Information Paths.

- **Tabla integrada**

Finalmente, se llevó a cabo el EDA de la tabla integrada, métricas GNN, Information Path, Network Proximity y Class, donde se verificó de nuevo la unicidad de los pares enfermedad-fármaco.

En esta tabla sí se observaron **valores nulos**, concretamente en las métricas de Network Proximity. No obstante, la proporción era moderada (entre un 1% y un 3%), siendo un porcentaje equilibrado en ambas clases. Por tanto, se optó por eliminar directamente las instancias con nulos, ya que el impacto sobre los datos era limitado y no justificaba la imputación, evitando así introducir posibles sesgos artificiales en las variables.

El **desequilibrio de clases** en la variable *Class* se mantuvo, perteneciendo ahora un 3.6162% de los pares (4,471 instancias) a la Clase 1, y el 96.3838% restante (119,167 instancias) a la Clase 0. Aunque esta proporción sigue siendo altamente desbalanceada, representa una mejora significativa frente al 0.0056% observado inicialmente. Este aumento de la clase positiva puede facilitar el entrenamiento del metamodelo, aunque se mantiene la necesidad de aplicar técnicas de balanceo a los datos para evitar un sesgo hacia la clase mayoritaria (Clase 0) durante el entrenamiento.

Los **boxplots** generados para las métricas mostraron valores atípicos en ambas clases. Sin embargo, estos *outliers* no fueron eliminados, ya que se encuentran en un rango razonable para cada una de las métricas. Esto ocurre debido a que la mayoría de los datos se agrupan en un rango muy pequeño, por lo que cualquier valor fuera de este rango se considera atípico.

Un aspecto destacable fue que, tras la integración de los datos, el patrón visto anteriormente en las métricas de las GNN, donde los pares de la Clase 1

(*treatment*) presentaban valores más altos sistemáticamente, dejó de reflejarse en los datos. En la tabla unificada, los valores de estas métricas son en general elevados, independientemente de la clase, lo que sugiere una **pérdida de la capacidad discriminativa aislada en estas variables**, lo que podría dificultar el aprendizaje del modelo.

Si nos centramos en el comportamiento de la **variable *path 2***, observamos un patrón atípico, tal y como se muestra en la **Figura 13**: en la clase *unknown* (0), todos los valores son 0, mientras que en la clase *treatment* (1), la mayoría toman el valor 1. Esta separación casi perfecta entre clases implica una alta correlación entre *path 2* y *class*.

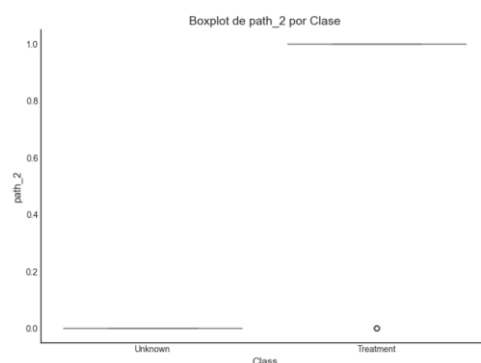


Figura 13. Boxplot de la variable *path 2* según la clase.

Mediante la **matriz de correlación de Spearman**, representada en la **Figura 14**, se verificó esta relación, obteniéndose un coeficiente de 0.98 entre ambas variables. Se determinó, gracias a este análisis, que lo más probable es que haya un error en el cálculo de la variable *path 2*: aparece siempre en aquellos pares enfermedad-fármaco que ya están reconocidos como tratamientos reales, debido a que no se aplicó un filtrado previo donde se excluyeran combinaciones clínicamente validadas en la actualidad.

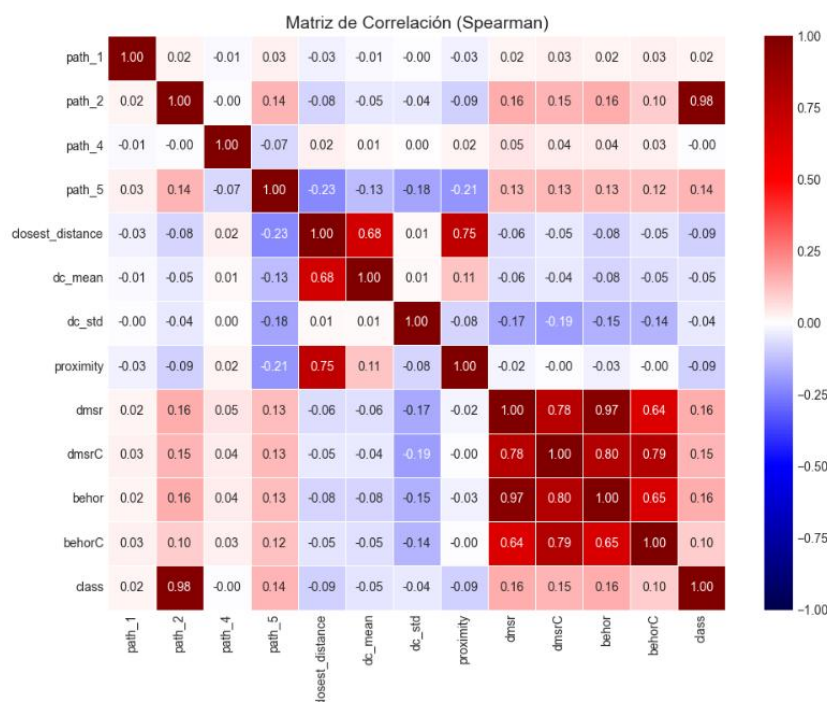


Figura 14. Matriz de correlación de Spearman entre las variables del *dataset* unificado.

Esto implica una contaminación del modelo, ya que *path 2* está codificando de forma directa la información que el modelo debería predecir, comprometiendo la separación entre datos de entrenamiento y prueba. Por tanto, se tomó la decisión de eliminar *path 2* del conjunto de características antes del entrenamiento del metamodelo, para evitar sesgos y garantizar una evaluación realista del rendimiento de este.

Por último, el análisis de correlación refuerza la **validez del enfoque integrador de este proyecto**, ya que no existe una correlación elevada entre métricas procedentes de modelos distintos, lo que indica que capturan información complementaria. Esta independencia entre fuentes heterogéneas de información apoya la hipótesis de que su combinación puede ayudar en la predicción de pares candidatos a DR.

Por otro lado, las métricas provenientes del mismo modelo (GNN o Network Proximity), presentan una correlación elevada entre sí, lo que indica que capturan información similar.

4.2 Estadísticas de la tabla final

El conjunto definitivo sobre el que se entrenaron los modelos de clasificación quedó constituido por 120,081 pares enfermedad-fármaco correspondientes a 116 enfermedades distintas y 1,069 fármacos únicos. Tras eliminar *path 2* y las instancias con valores nulos, la clase minoritaria (Clase 1) representa un 3.6% del total (4,345 instancias), frente al 96.4% de la Clase 0 (115,736 instancias), mostrando un desequilibrio todavía marcado. La **Tabla 8** recoge las estadísticas descriptivas correspondientes.

	Path 1	Path 4	Path 5	Closest distance	dc mean	dc std	Proximity	behor	behorC	dmsr	dmsrC	Class
Count	120081	120081	120081	120081	120081	120081	120081	120081	120081	120081	120081	120081
75%	0	1	0	2	2.015	0.392	0.767	1	1	1	1	0
50%	0	1	0	2	1.842	0.296	0.141	1	1	1	1	0
25%	0	1	0	1.5	1.630	0.211	-0.719	0.996	1	0.994	0.9998	0
Max	1	1	1	4	2.872	0.798	5.179	1	1	1	1	1
Mean	0.006	0.998	0.228	1.800	1.816	0.307	-0.017	0.930	0.98	0.908	0.955	0.036
Std	0.076	0.041	0.420	0.552	0.330	0.123	1.339	0.193	0.094	0.230	0.161	0.187
Min	0	0	0	0	0.121	0.016	-9.569	0.036	0.031	0.016	0.018	0

Tabla 8. Estadísticas descriptivas de la tabla integrada.

Las estadísticas descriptivas de la tabla, confirman las conclusiones extraídas durante el EDA:

- Las columnas binarias de **Information Paths** muestran la misma asimetría observada previamente: *path* 4 se activa en casi todo el conjunto (media ≈ 0.998), mientras que *path* 1 apenas aparece (0.6%). *Path* 5 mantiene un 23% de activación, lo que añade variabilidad útil.
- En las métricas de **Network Proximity** la mediana de *closest_distance* es 2, lo que implica que, para la mitad de los pares, los genes diana del fármaco se sitúan a tan solo dos saltos de los genes asociados a la enfermedad dentro de la red biomolecular; es, por tanto, una distancia corta que sugiere afinidad funcional. Al mismo tiempo, los valores de *proximity* oscilan entre -9.56 y 5.18 ($\sigma \approx 1.34$), indicando que esta métrica conserva capacidad discriminativa una vez integradas las tablas.
- Las variables de **GNN** (*behor*, *behorC*, *dmsr* y *dmsrC*) siguen muy concentradas en torno a 1 ($p_{25} = 0.99$; $p_{75} = 1$). Esta concentración de valores indica que, una vez integradas las tablas, aportan poca varianza relativa dentro de cada clase, por lo que el clasificador tendrá menos margen para explotar diferencias internas basadas únicamente en estas métricas.

Las distribuciones completas de todas las variables se presentan en el **Anexo II**, incluyendo histogramas en escala logarítmica para facilitar su visualización.

En resumen, se confirma que, aunque la fusión de fuentes ha reducido el desequilibrio extremo, la señal de ciertas variables (sobre todo las de la GNN) se han homogeneizado, reforzando así la decisión de combinar modelos heterogéneos y aplicar técnicas de balanceo. Así pues, el metamodelo deberá explotar las pequeñas diferencias locales, especialmente las aportadas por *Network Proximity* e *Information Paths*, para distinguir las escasas asociaciones verdaderas que hay en el conjunto de datos.

4.3 Evaluación de los Modelos

En esta fase de evaluación se diferencian varias etapas. En primer lugar, se comparan los distintos *pipelines*, evaluando su rendimiento tanto en validación cruzada (estabilidad *inter-fold*) como en los conjuntos de *test* externos, y seleccionando aquel que presente un mejor equilibrio entre ambos criterios. A

continuación, dentro del *pipeline* elegido, se contrastan los 5 métodos de clasificación para cada conjunto de *test*, seleccionando el que ofrezca la combinación más sólida de sensibilidad y precisión (*F1-score*). Finalmente, se analiza la calidad de las predicciones del método seleccionado, tanto de forma global como a nivel de *ranking*, con el fin de validar la efectividad farmacológica de los pares priorizados y detectar FP clínicamente relevantes.

4.3.1 Comparación de los *pipelines*

En este primer análisis de los resultados obtenidos, nos centramos en realizar una comparativa entre *pipelines*. La métrica guía fue el *F1-score* de la Clase 1, ya que en DR se prioriza identificar el mayor número posible de asociaciones terapéuticas, aun asumiendo cierto aumento de los falsos positivos.

- **Resultados de la validación cruzada anidada**

El siguiente aspecto a considerar es la **estabilidad *inter-fold***. Para evaluar la fiabilidad del modelo, se calculó en cada estrategia la media y la desviación estándar del *F1-class1* a lo largo de los cinco *folds* ($f1_mean \pm std$), procedimiento que se repitió para el resto de métricas. La **Figura 15** resume estos resultados para cada método de clasificación, mostrando el comportamiento promedio de cada *pipeline* durante la validación cruzada y permitiendo identificar posibles casos de sobreajuste o variabilidad excesiva entre *folds*.

Dicha figura muestra que *Oversampling* y *Downsampling* alcanzan valores en *F1-class1* muy elevados (cerca de 1), acompañados de sensibilidades (*recall*) y precisiones también extremas. El aparente liderazgo de *Oversampling* se basa en una fuga de información, reflejado en los valores elevados de las desviaciones estándar, ya que, al aplicar SMOTE antes de la validación cruzada, los ejemplos sintéticos creados durante el entrenamiento quedan replicados en los conjuntos de validación, de modo que el modelo distingue esos pares enfermedad-fármaco artificiales y no patrones genuinos. Al incluir SMOTE dentro de la validación cruzada, recreando los datos sintéticos de forma independiente en cada *fold*, los valores tan elevados desaparecen y el rendimiento se reduce hasta quedar al nivel del resto de *pipelines*, confirmando que la aparente mejora era falsa.

Downsampling muestra una situación diferente, ya que los promedios de las métricas son elevados, aunque las desviaciones se mantienen moderadas pero presentes, lo que sugiere un rendimiento interno estable y sin aparente sobreajuste. Por tanto, si nos centramos únicamente en los resultados de la validación cruzada, esta estrategia sería aparentemente más robusta. No obstante, la reducción considerable de la clase mayoritaria provocó un peor rendimiento en los conjuntos de *test*, evidenciando una menor capacidad de generalización, como se analizará a continuación.

Finalmente, los *pipelines* de *Cost sensitive* y *No Sampling* muestran medias más moderadas y desviaciones estándar contenidas, sin diferencias notables entre *folds*.

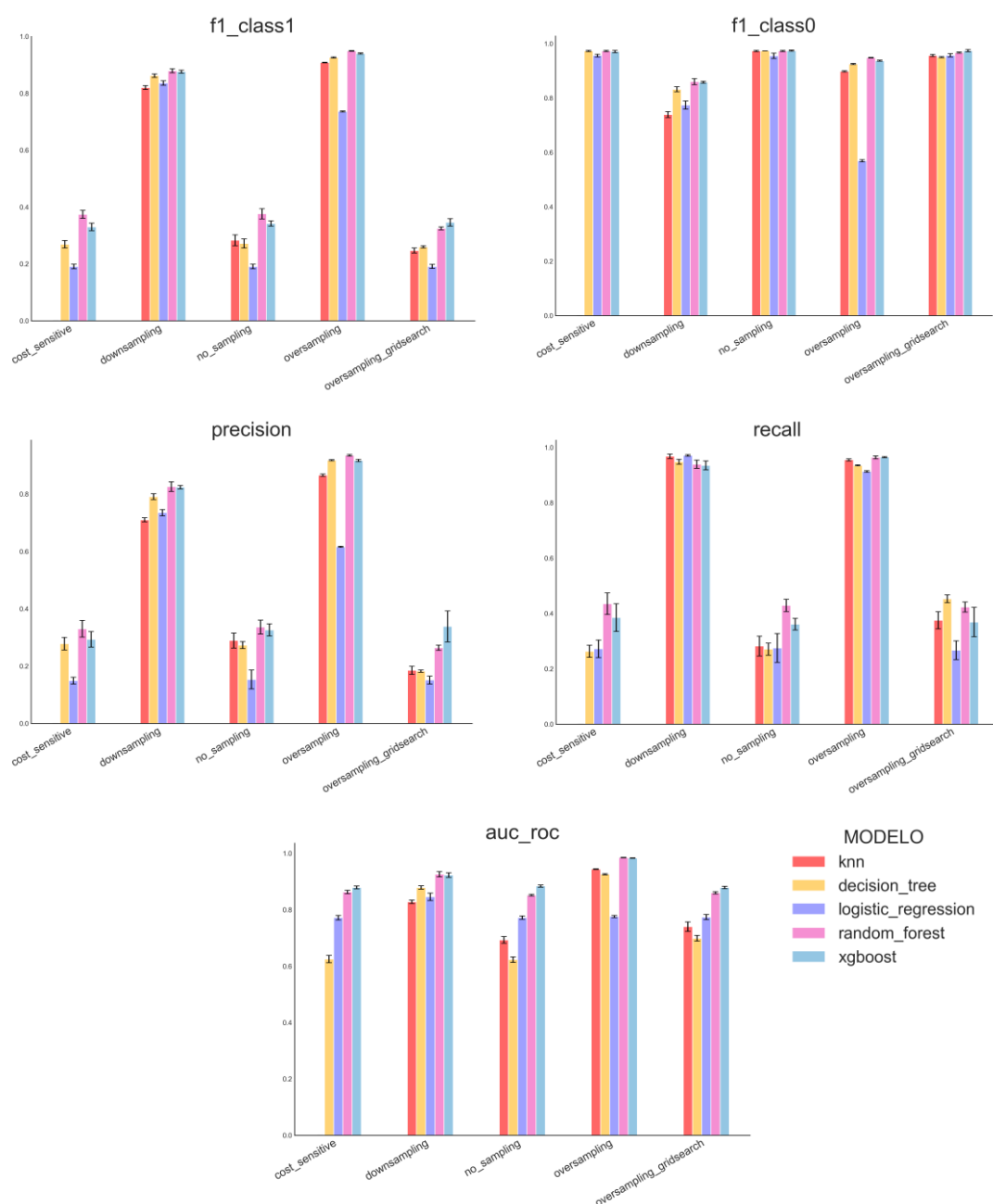


Figura 15. Comparación *inter-fold* de las métricas promedio para cada *pipeline* y métodos durante la validación cruzada.

Los resultados con las cifras numéricas exactas de la validación cruzada, incluyendo todas las métricas por *pipeline* y método, se recogen en el **Anexo III**.

- **Comprobación en los conjuntos de *test***

Los resultados detallados para cada combinación de pipelines y métodos se incluyen en la **Tabla 9** (alzhéimer) y la **Tabla 10** (esquizofrenia), donde se resumen las métricas clave calculadas sobre los conjuntos de *test* externos.

Al evaluar los distintos *pipelines* sobre los conjuntos de *test* externos, las diferencias tienden a estabilizarse. En el caso del alzhéimer, todas las métricas de *No Sampling* superan o igualan a sus correspondientes en las demás estrategias y obtienen, de forma consistente, *el F1-class 1* más elevado, manteniendo al mismo tiempo el equilibrio óptimo entre precisión y *recall* y con el AUC-ROC más alto. En esquizofrenia se observa el mismo patrón, donde los modelos entrenados sin técnicas de muestreo vuelven a presentar los valores más elevados en *recall* y AUC-ROC, mientras que su *F1-class 1* es bastante competitivo.

El *pipeline* de *Cost Sensitive* se aproxima a estas cifras, pero casi siempre con resultados ligeramente inferiores en las métricas principales. *Downsampling*, a pesar de su buen desempeño en la validación cruzada, no reproduce esos resultados en el *test* externo. Esto posiblemente se deba a que, al entrenar el modelo sobre una versión excesivamente simplificada y balanceada de la clase mayoritaria, este aprende patrones poco representativos que no es capaz de generalizar ante un conjunto de datos desbalanceado como el de *test*. Como consecuencia, se obtiene una precisión y *F1-score* por debajo de niveles clínicamente aceptables.

En *Oversampling*, como ya se explicó anteriormente, aplicar SMOTE antes de la validación cruzada provocó un incremento artificial de las métricas internas que no se reflejó en el *test* externo, donde se obtuvieron valores mínimos para precisión y *recall*. Sin embargo, cuando *Oversampling* se aplica dentro de la validación cruzada, eliminando así la fuga de información, su rendimiento se normaliza, pero sin superar al *pipeline No Sampling*. Además, aún siendo competitivo, sigue dependiendo de la generación de datos sintéticos, por lo que aunque en algunos modelos se obtienen resultados similares a las del enfoque sin muestreo, es preferible mantener la distribución natural de los datos y evitar introducir ejemplos artificiales que puedan provocar sesgos difíciles de detectar.

Pipeline	modelo	f1 class1	f1 class0	precision	recall	recall@100	auc_roc	TP	TN	FP	FN
Cost Sensitive	Decision Tree	0.381	0.981	0.364	0.4	0.5	0.733	12	999	21	18
	Logistic Regresion	0.222	0.965	0.167	0.333	0.4	0.775	10	970	50	20
	Random Forest	0.411	0.979	0.349	0.5	0.533	0.909	15	992	28	15
	XGBoost	0.429	0.98	0.375	0.5	0.867	0.964	15	995	25	15
Downsampling	Decision Tree	0.097	0.623	0.051	1.0	0.233	0.75	30	461	559	0
	KNN	0.019	0.948	0.013	0.033	0.067	0.532	1	945	75	29
	Logistic Regresion	0.0	0.986	0.0	0.0	0.133	0.579	0	1020	0	30
	Random Forest	0.116	0.71	0.061	1.0	0.267	0.837	30	562	458	0
	XGBoost	0.103	0.658	0.055	1.0	0.233	0.811	30	500	520	0
No Sampling	Decision Tree	0.333	0.976	0.286	0.4	0.4	0.686	12	990	30	18
	KNN	0.312	0.978	0.294	0.333	0.4	0.668	10	996	24	20
	Logistic Regresion	0.271	0.979	0.276	0.267	0.4	0.772	8	999	21	22
	Random Forest	0.405	0.978	0.341	0.5	0.6	0.897	15	991	29	15
	XGBoost	0.472	0.981	0.405	0.567	0.867	0.964	17	995	25	13
Oversampling	Decision Tree	0.175	0.942	0.112	0.4	0.4	0.653	12	925	95	18
	KNN	0.0	0.986	0.0	0.0	0.1	0.5	0	1020	0	30
	Logistic Regresion	0.057	0.984	0.2	0.033	0.167	0.759	1	1016	4	29
	Random Forest	0.254	0.952	0.167	0.533	0.533	0.887	16	940	80	14
	XGBoost	0.321	0.955	0.208	0.7	0.7	0.945	21	940	80	9
Oversampling Gridsearch	Decision Tree	0.248	0.951	0.162	0.533	0.533	0.739	16	937	83	14
	KNN	0.194	0.963	0.143	0.3	0.367	0.749	9	966	54	21
	Logistic Regresion	0.238	0.968	0.185	0.333	0.4	0.777	10	976	44	20
	Random Forest	0.308	0.973	0.25	0.4	0.6	0.892	12	984	36	18
	XGBoost	0.439	0.977	0.346	0.6	0.767	0.934	18	986	34	12

Tabla 9. Resultados de los métodos en los cinco *pipelines* para alzhéimer.

Pipeline	modelo	f1 class1	f1 class0	precision	recall	recall@100	auc_roc	TP	TN	FP	FN
Cost Sensitive	Decision Tree	0.263	0.932	0.215	0.338	0.324	0.635	23	888	84	45
	Logistic Regresion	0.125	0.12	0.067	0.956	0.368	0.783	65	62	910	3
	Random Forest	0.238	0.878	0.156	0.5	0.221	0.71	34	788	184	34
	XGBoost	0.235	0.913	0.172	0.368	0.294	0.724	25	852	120	43
Downsampling	Decision Tree	0.132	0.286	0.071	0.912	0.074	0.488	62	163	809	6
	KNN	0.092	0.641	0.052	0.397	0.059	0.427	27	478	494	41
	Logistic Regresion	0.0	0.966	0.0	0.0	0.029	0.349	0	972	0	68
	Random Forest	0.138	0.498	0.076	0.779	0.088	0.542	53	327	645	15
	XGBoost	0.143	0.418	0.078	0.882	0.118	0.58	60	259	713	8
No Sampling	Decision Tree	0.194	0.921	0.153	0.265	0.221	0.581	18	872	100	50
	KNN	0.19	0.929	0.16	0.235	0.235	0.641	16	888	84	52
	Logistic Regresion	0.14	0.344	0.076	0.926	0.368	0.784	63	203	769	5
	Random Forest	0.232	0.865	0.148	0.529	0.324	0.723	36	765	207	32
	XGBoost	0.244	0.899	0.169	0.441	0.294	0.742	30	824	148	38
Oversampling	Decision Tree	0.222	0.852	0.14	0.544	0.147	0.655	37	744	228	31
	KNN	0.0	0.966	0.0	0.0	0.074	0.5	0	972	0	68
	Logistic Regresion	0.043	0.955	0.08	0.029	0.235	0.722	2	949	23	66
	Random Forest	0.216	0.846	0.135	0.544	0.132	0.708	37	735	237	31
	XGBoost	0.195	0.843	0.122	0.485	0.221	0.722	33	734	238	35
Oversampling Gridsearch	Decision Tree	0.165	0.844	0.104	0.397	0.147	0.58	27	740	232	41
	KNN	0.166	0.909	0.124	0.25	0.176	0.595	17	852	120	51
	Logistic Regresion	0.124	0.018	0.066	1.0	0.368	0.783	68	9	963	0
	Random Forest	0.198	0.889	0.136	0.368	0.147	0.72	25	813	159	43
	XGBoost	0.213	0.9	0.15	0.368	0.221	0.725	25	830	142	43

Tabla 10. Resultados de los métodos en los cinco *pipelines* para esquizofrenia.

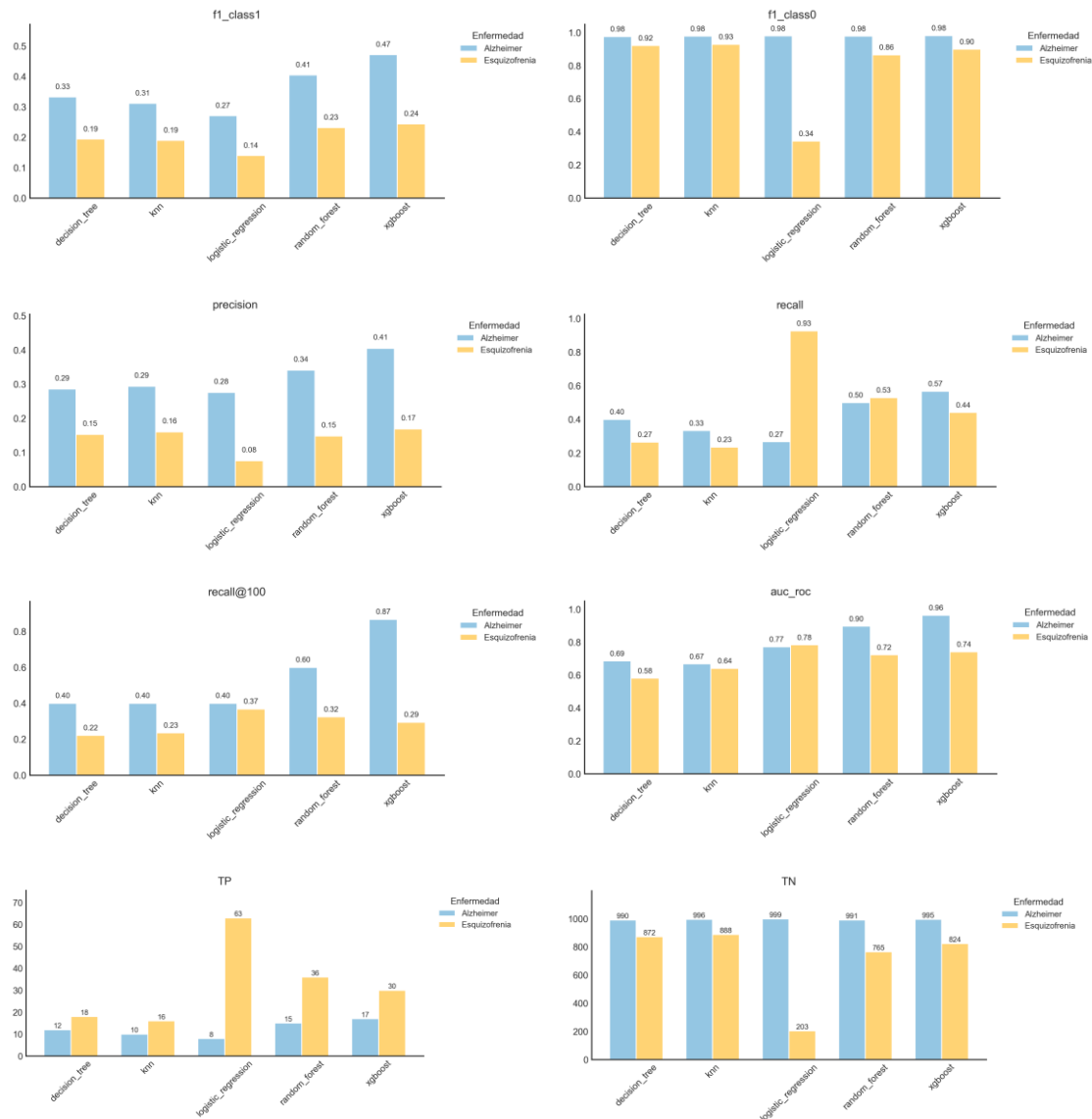
- **Interpretación y decisión final**

El comportamiento conjunto de la validación cruzada anidada y los resultados del *test* externo permite concluir que **No Sampling** es el *pipeline* que mejor armoniza sensibilidad clínica, precisión estadística y capacidad de generalización. A diferencia de *Oversampling*, que mostró una mejora artificial debido a la fuga de información introducida por SMOTE, y de *Downsampling*, cuyo excelente desempeño interno no se trasladó al *test*; No Sampling mantiene la distribución real de los datos y la menor discrepancia entre validación y prueba. Además, el coste computacional es menor que en las alternativas

basadas en balanceo o en búsqueda de pesos óptimos, lo que refuerza su elección como *pipeline* base.

4.3.2 Comparación de los métodos de clasificación en *No Sampling*

Tras seleccionar *No Sampling* como *pipeline* de referencia, se procede a comparar en detalle el comportamiento de los distintos métodos de clasificación sobre cada conjunto de *test*. La **Figura 16** muestra, para cada enfermedad y modelo, las métricas más relevantes: *F1-score* por clase, precisión, *recall*, *recall@100*, AUC-ROC, TP, TN, FP y FN.



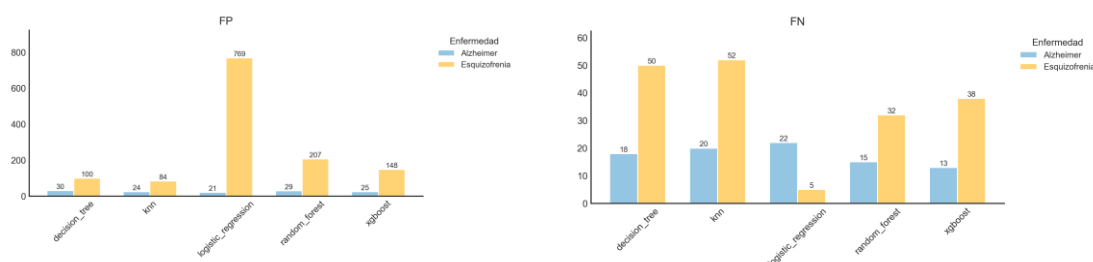


Figura 16. Comparativa de clasificadores sin muestreo en función de las métricas clave de evaluación, desglosadas por enfermedad.

En **alzhéimer** la superioridad es clara. XGBoost alcanza el *F1-class 1* más alto (0.472) y, al mismo tiempo, combina la mejor precisión (0.405) con el mayor *recall* (0.567); su *recall@100* asciende a 0.867 y la AUC-ROC se sitúa en 0.964, lo que evidencia una separación casi perfecta entre pares positivos y negativos. *Random Forest* aparece como segunda opción, aunque su F1 baja a 0.405 y la precisión a 0.341, mantiene un *recall* competitivo (0.5) y un AUC-ROC elevado (0.897). *Decision Tree*, KNN y Regresión Logística quedan en un segundo plano, con *F1-class 1* entre 0.271 y 0.333 y sin ninguna métrica que compense esa pérdida.

En **Esquizofrenia** la dispersión entre modelos se estrecha, pero la pauta general se mantiene. XGBoost vuelve a liderar el equilibrio entre sensibilidad y precisión: F1-class 1 de 0.244, *recall* de 0.441, precisión de 0.169 y AUC-ROC de 0.742. *Random Forest* ofrece un perfil muy parecido (F1 de 0.232 y *recall* de 0.529) pero con menor precisión y un AUC-ROC ligeramente inferior. Regresión Logística muestra un comportamiento inusual, el *recall* aumenta hasta 0.926 mientras que la precisión desciende a 0.076, lo que hunde el *F1-score* y multiplica los falsos positivos. Este desequilibrio está probablemente ligado a problemas de convergencia (se usó el valor *max_iter* = 100), es decir, el algoritmo no alcanza un óptimo estable y termina clasificando la mayoría de instancias como positivas para minimizar la función de pérdida, inflando así los FP y reduciendo el poder discriminativo del modelo. *Decision Tree* y KNN obtienen resultados modestos en todas las métricas.

El contraste entre ambas enfermedades refuerza la elección final. XGBoost es el método que se sitúa sistemáticamente en la primera posición, o muy cerca de ella, en todas las métricas clave y en ambas enfermedades, sin sacrificar de forma excesiva la precisión ni inflar el número de FP, y con un balance de TP/FN favorable. Por tanto, **XGBoost sobre el pipeline No Sampling se establece como clasificador de referencia**, mientras que *Random Forest* se conserva como alternativa viable, especialmente en escenarios similares al caso de esquizofrenia, donde presenta un mayor *recall* aunque a costa de incrementar el número de FP.

Las **Figuras 17** y la **Figura 18** complementan este análisis mostrando las curvas *recall@K* y *precision@K* para todos los modelos en *No Sampling*.

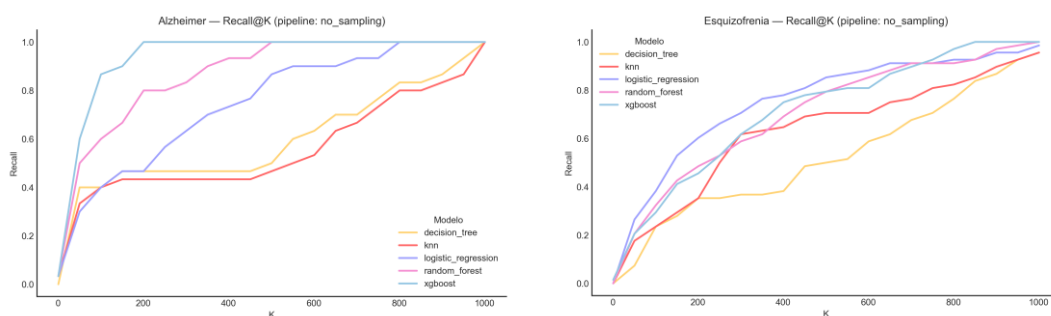


Figura 17. Evolución $Recall@K$ de los métodos de clasificación *No Sampling* en alzhéimer y esquizofrenia.

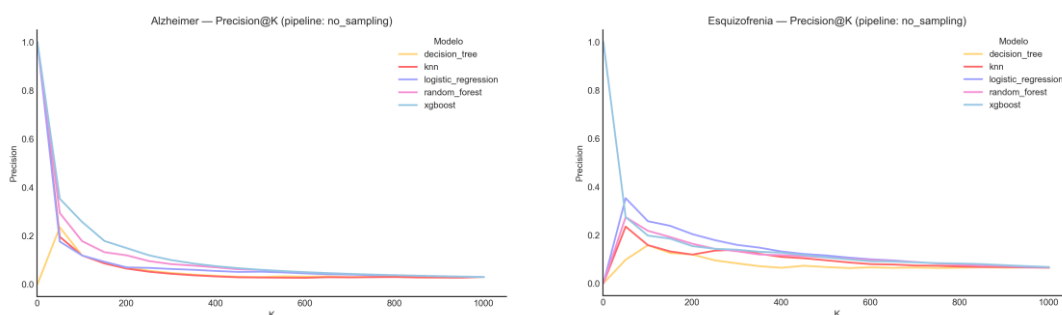


Figura 18. Evolución $Precision@K$ del de los métodos de clasificación *No Sampling* en alzhéimer y esquizofrenia.

Interpretar estas curvas en un escenario de trabajo real resulta clave, ya que en el contexto de DR, rara vez se revisan miles de pares enfermedad-fármaco, sino que suelen centrarse en los primeros 100 candidatos del ranking, lo que correspondería a $recall@100$.

Debe recordarse, además, que los conjuntos de prueba contienen 30 pares positivos en alzhéimer y 68 en esquizofrenia, por lo que probablemente, una vez superado ese umbral (valor de k), el $recall$ difícilmente seguirá aumentando, mientras que la precisión tenderá a disminuir, ya que los nuevos elementos incorporados al *ranking* serán, en su mayoría, negativos por definición.

En **$recall@K$** , las curvas presentan dos dinámicas distintas. En alzhéimer, XGBoost asciende casi verticalmente alcanzando un $recall@100$ de 0.867, lo que equivale a identificar aproximadamente el 87% de los tratamientos verdaderos entre los 100 primeros resultados (26 de los 30 tratamientos reales), destacando por encima del resto de modelos. En esquizofrenia, en cambio, Regresión Logística comienza con la pendiente más pronunciada y lidera el tramo inicial ($recall@100 = 0.368$ frente al 0.294 de XGBoost, es decir, 25 y 20 tratamientos identificados de 68), pero a partir de $K = 700$ XGBoost termina alcanzando antes el $recall$ máximo. Esta diferencia entre ambas enfermedades sugiere que en alzhéimer los modelos son capaces de concentrar más rápidamente los pares positivos al inicio del ranking, mientras que, en esquizofrenia, por el contrario,

los TP se distribuyen de forma más gradual, obligando a recorrer un mayor número de candidatos para identificarlos.

En **precision@K** la situación cambia. En alzhéimer, XGBoost presenta la mayor precisión a lo largo de toda la curva, estabilizándose cerca de 0.18 hasta $K = 250$, lo que permite ampliar la lista de candidatos sin que la proporción de FP aumente de manera extrema. En esquizofrenia, las precisiones iniciales también están encabezadas por XGBoost, ya que se mantiene por encima de los demás hasta $K = 40$, pero a partir de ese punto la Regresión Logística presenta valores más elevados: entre $K = 50$ y $K = 300$ su curva se alza por encima de las demás, reflejo de que, aun con un gran volumen global de FP (concretamente 769 instancias), concentra un número relativamente alto de TP (63 instancias de 68) en ese tramo intermedio del *ranking*. Sin embargo, estos valores más elevados de precisión no compensan el impacto de los errores acumulados, de modo que el rendimiento general del modelo continúa siendo inferior al del resto.

Para cerrar el apartado de resultados, la **Figura 19** y la **Figura 20** muestran la curva ROC y la matriz de confusión de XGBoost para ambas enfermedades.

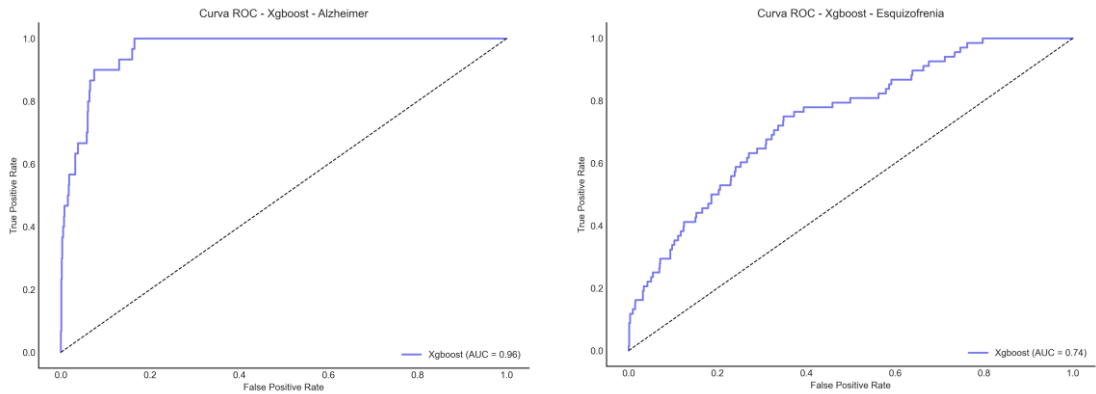


Figura 19. Curva AUC-ROC de XGBoost en alzhéimer y esquizofrenia.

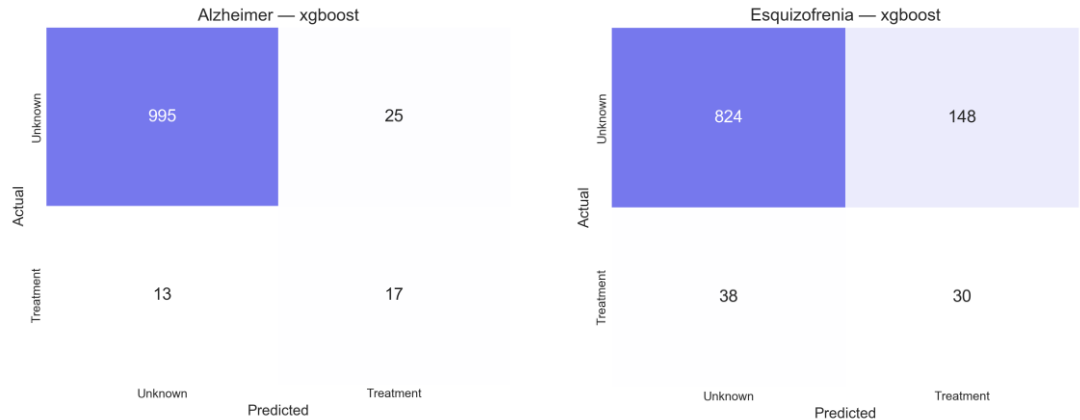


Figura 20. Matriz de Confusión de XGBoost en alzhéimer y esquizofrenia.

En **alzhéimer**, XGBoost alcanza un AUC de 0.96, reflejo de una capacidad casi perfecta para distinguir pares *treatment* y *unknown*. Con el umbral óptimo

elegido se obtienen 17 TP frente a 13 FP, es decir, identifica más de la mitad de los 30 tratamientos reales, y solo 25 FP entre 1,050 pares totales, de modo que la carga experimental se mantiene dentro de márgenes asumibles.

En **esquizofrenia**, la discriminación de clases es menor ($AUC = 0.74$). La matriz de confusión confirma el descenso de rendimiento, 30 TP y 38 FP, junto con 148 FP. Así, el modelo recupera alrededor del 44% de los 68 tratamientos reales, pero a costa de aumentar el número de candidatos irrelevantes. El contraste entre ambas curvas ROC y el notable aumento de FP reflejan que el método de XGBoost ofrece un comportamiento robusto en alzhéimer, mientras que en esquizofrenia su capacidad discriminativa se reduce y exige un esfuerzo de validación experimental mucho mayor.

4.3.3 Comparación global entre alzhéimer y esquizofrenia

Un último aspecto de interés es la comparación global entre las dos enfermedades de *test*. En **alzhéimer**, el modelo parece integrar mejor la información, generando predicciones más coherentes y estables. Esto podría deberse a una mayor alineación entre las métricas de entrada, con una señal menos ruidosa, lo que facilitaría que el algoritmo discrimine entre pares *treatment* y *unknown*.

En **esquizofrenia**, por el contrario, la señal procedente de las métricas es más heterogénea, lo que dificulta la separación entre clases. Además, probablemente al estar formado el conjunto de *test* con una proporción mayor de pares positivos, aumenta la probabilidad de confundir patrones y generar falsos positivos. Este comportamiento podría estar también relacionado con las propias limitaciones de los tratamientos actuales en esquizofrenia. Aunque existen múltiples fármacos aprobados, su efectividad no es universal ni cubre de forma completa todos los síntomas del trastorno. Por tanto, el modelo puede estar identificando como positivos algunos compuestos aún no utilizados, proponiendo nuevas hipótesis de reposicionamiento que merecen evaluación experimental [49].

Por tanto, esta comparación señala la importancia de adaptar la estrategia de aprendizaje automático al contexto específico de cada enfermedad y de planificar la validación experimental posterior en consecuencia. En patologías donde las métricas de entrada reflejan patrones más claros y consistentes, como ocurre en alzhéimer, modelos como XGBoost pueden generar clasificaciones más fiables. Sin embargo, en enfermedades como la esquizofrenia, este tipo de algoritmos podría requerir ajustes adicionales para mantener un balance aceptable entre sensibilidad y carga experimental.

4.3.4 Resultados de las predicciones

Por último, se comparan las predicciones generadas por XGBoost para ambas enfermedades con las etiquetas reales de clase. La **Tabla 11** y la **Tabla 12**, recogen estos resultados para cada enfermedad.

Para valorar el potencial del metamodelo, resulta relevante examinar los falsos positivos, ya que representan hipótesis de reposicionamiento aún no validadas clínicamente. Por su parte, los verdaderos positivos actúan como control interno del modelo, al corresponderse con pares previamente aprobados como

tratamientos, lo que permite comprobar que el sistema es capaz de identificar compuestos conocidos y efectivos.

Disease_id	Drug_name	y_proba	y_pred	class
C0002395	MELATONIN	0.896596	1	1
C0002395	NICOTINE	0.881311	1	1
C0002395	NIMESULIDE	0.878985	1	0
C0002395	TACRINE	0.874289	1	1
C0002395	GALANTAMINE	0.863449	1	1
C0002395	IBUPROFEN	0.809022	1	1
C0002395	RIVASTIGMINE	0.753526	1	1
C0002395	MINOCYCLINE	0.741506	1	1
C0002395	TADALAFIL	0.680720	1	0
C0002395	PINDOLOL	0.628993	1	1

Tabla 11. Predicciones de XGBoost en alzhéimer.

En alzhéimer, entre los diez primeros pares priorizados por el modelo aparecen dos FP: **nimesulida** (un AntiInflamatorio No Esteroideo, AINE) y **tadalafil** (inhibidor PDE-5). Aunque ambos figuran en la tabla como Clase 0, el modelo los etiqueta como positivos porque su probabilidad supera el umbral óptimo. Es posible que estas predicciones no sean arbitrarias, ya que existen estudios que analizan los AINE [50] y los fármacos vasodilatadores [51] como posibles protectores frente a demencia, aunque sin aprobación clínica todavía. Por tanto, estos pares enfermedad-fármaco pueden interpretarse como propuestas razonables de reposicionamiento que merecen ser exploradas.

Junto a estos pares, el modelo identifica también en las primeras posiciones numerosos compuestos con sólida evidencia bibliográfica en el tratamiento de la demencia, lo que demuestra su capacidad para recuperar tratamientos ya validados. **Tacrina** fue el primer inhibidor de colinesterasa aprobado para la enfermedad [52], **galantamina y rivastigmina** forman parte de la terapia estándar [53] [54], **melatonina e ibuprofeno** cuentan con estudios que apuntan a beneficios sintomáticos o preventivos [55] [56], **minociclina** se ha ensayado como modulador inflamatorio [57], y la **nicotina**, aunque no es un fármaco per se, se ha investigado como agonista colinérgico de acción rápida [58].

Disease_id	Drug_name	y_proba	y_pred	class
C0036341	LEVOMEPROMAZINE	0.982847	1	1
C0036341	CABERGOLINE	0.979993	1	0
C0036341	AMITRIPTYLINE	0.978739	1	1
C0036341	CHLORPROMAZINE	0.977460	1	1
C0036341	QUETIAPINE	0.970127	1	1
C0036341	CLOZAPINE	0.969319	1	1
C0036341	LOXAPINE	0.959326	1	1
C0036341	ESTRADIOL BENZOATE	0.954857	1	0
C0036341	PSEUDOEPHEDRINE	0.946559	1	0
C0036341	OLANZAPINE	0.945407	1	1

Tabla 12. Predicciones XGBoost esquizofrenia.

En **esquizofrenia**, los FP priorizados por el modelo muestran una perspectiva relevante sobre sus limitaciones y oportunidades. Entre los primeros pares aparecen cabergolina, estradiol benzoato y pseudoefedrina, clasificados como Clase 0 pero etiquetados por el metamodelo como Clase 1.

Cabergolina, se usa como coadyuvante para revertir la hiperprolactinemia inducida por antipsicóticos, con cierta mejoría endocrina pero riesgo de reactivar la psicosis si la dosis no se ajusta con cautela, lo que la convierte en una propuesta de reposicionamiento con potencial clínico pero que requiere vigilancia estrecha [59]. **Estradiol benzoato**, ha sido evaluado como tratamiento complementario en mujeres con esquizofrenia, sobre todo en mujeres premenopáusicas, y han informado reducciones moderadas de síntomas positivos, aunque su empleo requiere monitorizar efectos secundarios propios de la terapia estrogénica [60].

Por el contrario, **pseudoefedrina** carece de evidencia terapéutica en esquizofrenia y, de hecho, se ha documentado como posible desencadenante de brotes psicóticos en pacientes vulnerables [61]. Así pues, su aparición como FP subraya un riesgo claro y evidencia una limitación fundamental del metamodelo. Estos errores podrían explicarse en parte por el tipo de información utilizada para entrenar el modelo, ya que las métricas de entrada derivan de grafos sin relaciones causales explícitas, por lo que el modelo no es capaz de distinguir si un compuesto se relaciona con una mejora o con un empeoramiento del estado clínico, es decir, si refleja un efecto terapéutico o adverso.

Pese a ello, el modelo también coloca en las primeras posiciones múltiples tratamientos ya aprobados y empleados en la práctica, **levomepromazina**, **clorpromazina**, **quetiapina**, **clozapina**, **loxapina**, **olanzapina** y **amitriptilina**. Todos ellos figuran correctamente como Clase 1 y actúan como antipsicóticos o moduladores de síntomas negativos [62] [63].

4.4. Comparación con Crank

Finalmente, se llevó a cabo la comparación externa con *crank score*, con el fin de contrastar las probabilidades generadas por XGBoost con un *ranking* construido exclusivamente a partir de las 11 métricas de entrada del metamodelo. El procedimiento llevado a cabo para construir este *ranking* fue descrito en la **subsección 3.2.5.2**.

Para evaluar la coherencia entre ambos enfoques, se seleccionaron los pares que XGBoost clasificó como positivos (ordenados por su probabilidad), y se examinó el *crank score* asignado a cada uno. Los primeros resultados para alzhéimer y esquizofrenia se recogen, respectivamente, en la **Tabla 13** y en la **Tabla 14**.

Disease_id	Drug_name	y_proba	y_pred	class	Crank_score
C0002395	MELATONIN	0.896596	1	1	0.943550
C0002395	NICOTINE	0.881311	1	1	0.887158
C0002395	NIMESULIDE	0.878985	1	0	0.735933
C0002395	TACRINE	0.874289	1	1	0.889286
C0002395	GALANTAMINE	0.863449	1	1	0.910645
C0002395	IBUPROFEN	0.809022	1	1	0.905440
C0002395	RIVASTIGMINE	0.753526	1	1	0.887167
C0002395	MINOCYCLINE	0.741506	1	1	0.954824
C0002395	TADALAFIL	0.680720	1	0	0.692082
C0002395	PINDOLOL	0.628993	1	1	0.836586

Tabla 13. Comparación resultados XGBoost frente a *crank score* para alzhéimer.

Disease_id	Drug_name	y_proba	y_pred	class	Crank_score
C0036341	LEVOMEPROMAZINE	0.982847	1	1	0.986473
C0036341	CABERGOLINE	0.979993	1	0	0.987085
C0036341	AMITRIPTYLINE	0.978739	1	1	0.986802
C0036341	CHLORPROMAZINE	0.977460	1	1	0.987993
C0036341	QUETIAPINE	0.970127	1	1	0.990200
C0036341	CLOZAPINE	0.969319	1	1	0.990963
C0036341	LOXAPINE	0.959326	1	1	0.987939
C0036341	ESTRADIOL BENZOATE	0.954857	1	0	0.966452
C0036341	PSEUDOEPHEDRINE	0.946559	1	0	0.991501
C0036341	OLANZAPINE	0.945407	1	1	0.987429

Tabla 14. Comparación resultados XGBoost frente a *crank score* para esquizofrenia.

En alzhéimer, el *crank score* corrobora en gran medida la priorización de XGBoost: todos los tratamientos aprobados (Clase 1) situados en los primeros puestos del *ranking* superan la barrera de 0.83. Resulta llamativo, que los dos FP del modelo, nimesulida y tadalafilo, reciban puntuaciones sensiblemente más bajas en *crank* (0.74 y 0.69), lo que sugiere que el agregado lineal de métricas sí detecta una señal diferenciadora que el clasificador no captó. En cambio, esto no se cumple para esquizofrenia, donde tanto los TP como los tres FP obtienen valores muy altos, superiores incluso a 0.96.

El comportamiento global de *crank* se observa con mayor claridad en la **Figura 21**, donde se representa el número de pares con *crank score* > 0.9 desglosado por clase y enfermedad. En alzheimer, apenas 24 pares superan ese umbral; 19 pertenecen a la Clase 0 y tan solo 5 a la Clase 1. En esquizofrenia la situación se desborda, ya que el umbral agrupa 301 pares, de los cuales 256 son Clase 0 y solo 45 Clase 1. Es decir, en un escenario de señal molecular heterogénea, *crank* tiende a inflar la lista con un volumen considerable de candidatos potencialmente irrelevantes.

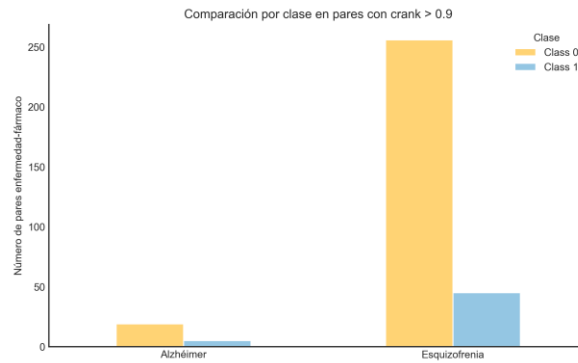


Figura 21. Distribución de pares con *crank score* > 0.9 por clase para alzhéimer y esquizofrenia.

Estos resultados deben interpretarse teniendo en cuenta los resultados de la matriz de correlación mostrada en la **Figura 14**, donde ninguna de las métricas, salvo *path 2* que fue eliminada, presentaba correlación con la variable objetivo (*class*). En este contexto, basar el *crank score* en una agregación puramente lineal de dichas métricas resulta metodológicamente cuestionable. En alzhéimer, donde la señal biológica está mejor definida, la suma ponderada consigue distinguir en cierta medida los pares positivos. Sin embargo, en esquizofrenia, la falta de correlación se traduce en un *ranking* saturado de FP. Por tanto, *crank* ofrece una validación complementaria pero muy condicionada por la calidad de la señal. Al tratarse de un enfoque estadístico no supervisado basado en combinaciones lineales, su capacidad para detectar interacciones complejas entre métricas es limitada, lo que refuerza la utilidad de emplear métodos de *machine learning* supervisados, más expresivos y adaptativos a patrones no lineales.

5 Conclusiones y líneas futuras

5.1 Conclusiones

La realización de este TFG ha demostrado que la **integración de enfoques computacionales heterogéneos mediante un metamodelo de machine learning** es una estrategia válida y eficaz para priorizar hipótesis de reposicionamiento de fármacos. Partiendo de la identificación de la falta de integración entre los distintos modelos almacenados en DRIVE, se construyó un conjunto de datos unificado que combina métricas procedentes de GNNs, *Network Proximity* e *Information Paths*, junto con la variable objetivo derivada de *disease_drug*. El análisis exploratorio confirmó la independencia de los modelos computacionales de DR y evidenció cuestiones técnicas clave; el fuerte desequilibrio de clases y la fuga de información asociada a *path 2*, que hubieran sesgado cualquier conclusión si no se hubiesen detectado a tiempo.

Tras evaluar cinco estrategias de muestreo y cinco clasificadores (KNN, *Decision Tree*, *Random Forest*, Regresión Logística y XGBoost) mediante validación cruzada anidada, el **pipeline sin técnicas de balanceo** se mostró como la opción más robusta, pues evita el sesgo que introduce la generación de datos sintéticos o la eliminación excesiva de ejemplos reales y, al mismo tiempo, mantiene una varianza *inter-fold* reducida junto con un rendimiento competitivo en los conjuntos de *test*. Dentro de este *pipeline*, **XGBoost** ofreció el equilibrio más consistente entre sensibilidad (*recall*) y precisión, es decir, para el *F1-score*, tanto en alzhéimer como en esquizofrenia.

No obstante, el rendimiento no fue uniforme en los dos conjuntos de prueba. En **alzhéimer** el modelo recuperó alrededor del 87% de los tratamientos aprobados dentro de los 100 primeros candidatos, y los pocos FP se correspondieron con moléculas que ya cuentan con estudios preclínicos o hipótesis razonables de actividad neuroprotectora. En **esquizofrenia**, en cambio, la mayor heterogeneidad de los datos se tradujo en un aumento de FP, incluidos fármacos capaces de exacerbar la psicosis, lo que evidencia que, cuando los datos de entrada son difusos o inconsistentes, el metamodelo pierde capacidad discriminativa y exige una validación experimental más rigurosa. Además, los resultados han estado condicionados por el fuerte desequilibrio de clases, un reto asumido desde el inicio y que sigue limitando la capacidad discriminativa del modelo, especialmente en patologías con señal molecular difusa.

Finalmente, el contraste con *crank score* mostró las limitaciones de este enfoque, ya que al basarse en una agregación estrictamente lineal de métricas que no presentan correlación con la variable objetivo, el *ranking* generado resulta poco fiable. Aunque en alzhéimer la puntuación agregada coincide con la priorización de XGBoost y penaliza los dos FP, en esquizofrenia asigna valores muy elevados de forma indiscriminada a pares de ambas clases, incluyendo un gran número de FP. Esto pone en duda su utilidad como referencia externa y refuerza la necesidad de emplear métodos supervisados de *machine learning*, capaces de capturar relaciones más complejas, principalmente si la señal biológica es difusa.

El proyecto, por tanto, confirma la hipótesis de partida, **una capa de ML puede sintetizar métricas heterogéneas y producir rankings coherentes**, pero también subraya sus límites. El metamodelo ofrece una base priorizada sobre la que focalizar recursos experimentales más costosos (ensayos *in vitro* e *in vivo*), reduce el espacio de búsqueda y aporta trazabilidad reproducible gracias a su código abierto y estructura modular; sin embargo, no reemplaza el juicio clínico ni el cribado experimental.

La fiabilidad real de las predicciones depende de la calidad y la homogeneidad de los datos disponibles para cada enfermedad, de modo que deben interpretarse como una guía preliminar y no como una respuesta definitiva. Solo la colaboración conjunta entre modelos *in silico* (modelos computacionales), cada vez más expresivos y evidencias experimentales de mayor calidad hará posible que esta aceleración teórica se traduzca en beneficios terapéuticos tangibles, reduciendo de forma realista costes y tiempos frente al desarrollo *de novo* sin sacrificar la seguridad ni la eficacia clínica.

5.2 Líneas Futuras

Los resultados obtenidos confirman la utilidad del metamodelo para priorizar hipótesis de reposicionamiento, pero también señalan varias líneas futuras que permitirán fortalecer y ampliar el alcance de este TFG.

La primera de ellas es **ampliar la búsqueda de hiperparámetros**. En este proyecto se exploraron opciones moderadas para cada clasificador, debido a las limitaciones de cómputo, pero una búsqueda más exhaustiva (con enfoques más complejos o con un mayor número de combinaciones de parámetros a evaluar) podría mejorar de forma apreciable la capacidad predictiva del modelo. Para ello sería necesario ejecutar los experimentos en un entorno con mayor capacidad computacional, conectándose a los servidores del laboratorio o empleando infraestructuras de cómputo distribuido en la nube.

Una segunda línea futura consiste en **aumentar el conjunto de datos de entrada del metamodelo**, incorporando las tablas disponibles en DRIVE que no fueron utilizadas en este proyecto. Hasta ahora se han integrado métricas procedentes de tres modelos computacionales (`dr_gnns`, `dr_network_proximity` y `dr_information_paths`), que han demostrado aportar señales complementarias y relevantes para la predicción de candidatos a reposicionamiento. Sin embargo, tablas como `dr_pathways`, `dr_diseasepathways` o `dr_threshold_values`, contienen información sobre rutas biológicas, recuentos de implicación funcional y umbrales de decisión derivados de estudios integradores.

Su incorporación permitiría analizar el efecto de la multifaceticidad de los datos, es decir, evaluar si la combinación de diferentes dimensiones del conocimiento biomédico mejora la capacidad predictiva y la robustez del metamodelo, o si, por el contrario, introduce redundancias o ruido que exija aplicar técnicas de selección de características o reducción de dimensionalidad.

Una tercera línea futura se orienta a **reducir la brecha entre predicción algorítmica y utilidad clínica**, mediante el desarrollo de métodos de estimación de incertidumbre y su integración con técnicas de interpretabilidad como SHAP o LIME. Esta combinación permitiría que cada predicción vaya acompañada tanto de la aportación relativa de cada métrica al resultado como

de una estimación del grado de confianza asociado a la recomendación. Esto reforzaría la transparencia del sistema y facilitaría que los expertos clínicos evalúen la solidez de los candidatos antes de avanzar hacia su validación experimental.

En resumen, las líneas de trabajo propuestas combinan un refinamiento metodológico con una orientación a la aplicación clínica. Este doble enfoque, que fortalece tanto el rendimiento algorítmico como la transparencia del proceso, posiciona al metamodelo como una herramienta fiable y explicable para priorizar los fármacos con mayor potencial de reposicionamiento.

6 Análisis de Impacto

En este capítulo se analiza el impacto potencial del TFG desde diferentes perspectivas: **personal, empresarial, social, económica, medioambiental y cultural**. Además, se examina su contribución a los **Objetivos de Desarrollo Sostenible (ODS)** establecidos en la Agenda 2030 [64].

Con este análisis identificaremos los beneficios del metamodelo propuesto, así como los posibles efectos perjudiciales que puedan derivarse de su uso. En cada una de estas perspectivas, se valoran tanto los aportes directos como los riesgos asociados, junto con las decisiones llevadas a cabo a lo largo del proyecto para potenciar su impacto positivo y mitigar los efectos adversos.

6.1. Impacto personal

Desde la perspectiva personal, este TFG ha supuesto una gran oportunidad de aprendizaje. A nivel técnico, me ha permitido afianzar y desarrollar mis conocimientos en ML, aplicándolos a un contexto biomédico real, concretamente al reposicionamiento de fármacos, un área que hasta ahora no había explorado en detalle. Gracias al acceso a los datos heterogéneos de la plataforma DISNET, he podido aplicar mi experiencia previa en análisis de datos con Python —especialmente con la librería Pandas— al tiempo que desarrollaba nuevas competencias en el uso de herramientas como Dask, muy importantes para el tratamiento eficiente de grandes volúmenes de datos. Además, el trabajo ha servido para reforzar conocimientos adquiridos anteriormente sobre bases de datos relacionales, especialmente MySQL, adaptados ahora a un entorno de alta carga y consulta masiva de datos.

Todo ello ha favorecido también el desarrollo de habilidades personales clave. La **autonomía en la investigación** fue especialmente importante al inicio del proyecto, durante la fase de revisión del estado del arte, que requirió buscar, leer y entender distintos enfoques de DR. La **planificación a largo plazo** resultó esencial para organizar un proyecto con múltiples etapas a lo largo de varios meses, evitando retrasos y permitiendo una evolución constante. Además, la **capacidad de análisis crítico** fue necesaria para tomar decisiones metodológicas relevantes, como la selección de métricas, algoritmos o estrategias de preprocesado. Finalmente, se afianzaron buenas prácticas de **ingeniería de datos y reproducibilidad**, asegurando la trazabilidad del código, el uso de formatos eficientes y una estructura clara del flujo de trabajo.

Por otro lado, al tratarse de un proyecto extenso y con un alto nivel de implicación personal, existe el riesgo de caer en una sobreinversión de tiempo y sentirse agobiado con sensación de bloqueo. No obstante, estos retos también han sido clave para aprender a gestionar mejor el tiempo y mantener el equilibrio con otras responsabilidades académicas.

6.2. Impacto empresarial

El metamodelo propuesto responde a necesidades reales de la industria farmacéutica y biotecnológica: acelerar las fases iniciales del descubrimiento de fármacos y reducir el riesgo financiero, tiempo y coste de desarrollo de nuevos fármacos. Al integrar los resultados generados por tres modelos

computacionales distintos, GNNs, *Network Proximity* e *Information Paths*, el sistema facilita una toma de decisiones basada en datos.

Asimismo, los resultados obtenidos a través del metamodelo, es decir, la priorización de pares enfermedad-fármaco con mayor probabilidad de éxito, podrían ser utilizados por terceros, como *start-ups* de health-tech, departamentos de I+D o plataformas académicas, como punto de partida en el proceso de DR. A partir de estas predicciones, podrían iniciarse etapas posteriores de validación experimental, ensayos preclínicos o análisis farmacológicos más detallados. De este modo, el proyecto no solo aporta una herramienta de apoyo a la toma de decisiones, sino que también contribuye a establecer un puente entre la investigación académica y la industria farmacéutica.

No obstante, una confianza excesiva en las predicciones podría llevar a invertir recursos en candidatos que, pese a la señal computacional generada por el metamodelo, fracasen clínicamente.

6.3. Impacto social

El impacto social del proyecto se basa en su capacidad para facilitar el acceso a tratamientos más rápidos y asequibles, lo que mejoraría la equidad y reduciría las desigualdades sociales en salud. Resulta especialmente interesante en el caso de enfermedades raras o desatendidas, donde la baja rentabilidad limita la inversión industrial. Al reutilizar fármacos ya aprobados y seguros, el DR reduce costes y tiempos de desarrollo, esto contribuye a la optimización de los recursos sanitarios y permite liberar recursos para otras necesidades sociales. En este contexto, el metamodelo propuesto puede servir como herramienta de apoyo para priorizar combinaciones con mayor probabilidad de éxito, acelerando la generación de hipótesis terapéuticas viables y ampliando las opciones disponibles para estos pacientes.

Al reducir tiempos y costes, se incrementa la probabilidad de que hospitales, fundaciones o iniciativas de salud pública adopten estas hipótesis terapéuticas y las pongan a prueba para mejorar la salud y la calidad de vida. Concretamente, este metamodelo se ha querido probar con dos enfermedades con un alto impacto social en la actualidad como son la esquizofrenia y el alzhéimer.

No obstante, se reconoce la necesidad de evitar una divulgación inadecuada de los resultados que pudiera generar expectativas infundadas. Por ello, a lo largo del proyecto se ha insistido en la importancia de contextualizar el modelo como una herramienta de apoyo a la toma de decisiones, no como un sustituto del juicio clínico ni de la validación experimental.

Por último, hay que señalar que cuando se asigne una nueva indicación terapéutica a un fármaco ya comercializado, la industria farmacéutica debe de prever la mayor demanda que va a tener en el mercado y evitar situaciones de desabastecimiento, que irían en detrimento de los tratamientos ya establecidos.

6.4. Impacto económico

Desde el punto de vista económico, el modelo contribuye a reducir significativamente los costes asociados al desarrollo de nuevos tratamientos. El enfoque tradicional de fármacos implica inversiones que pueden superar los

2.000 millones de dólares y plazos de entre 10 y 17 años. El reposicionamiento, en cambio, aprovecha compuestos ya aprobados, disminuyendo el riesgo toxicológico y acortando el tiempo de desarrollo a 3-12 años.

El metamodelo desarrollado en este proyecto apoya esta estrategia al priorizar de forma eficiente las combinaciones más prometedoras entre pares enfermedad y fármaco, es decir, se evita destinar fondos a hipótesis con señal débil o redundante y se concentran esfuerzos en aquellas con respaldo convergente de varios modelos.

El DR permite a la industria farmacéutica obtener beneficios con una inversión menor, a la vez que abarata los precios y disminuye el gasto farmacéutico en salud. Esta mayor rentabilidad del DR, en ningún caso debe ser un factor limitante de la inversión que, tanto la industria farmacéutica como los entes públicos, dedican a la investigación tradicional de nuevas moléculas y alternativas terapéuticas.

6.5. Impacto medioambiental

Aunque en primera instancia el impacto medioambiental de un sistema computacional puede parecer reducido, el modelo desarrollado contribuye a una reducción indirecta del consumo de recursos en etapas preclínicas y experimentales. Al mejorar la capacidad de priorización de candidatos terapéuticos, se reduce la necesidad de realizar ensayos presenciales innecesarios, disminuyendo así el uso de materiales de laboratorio y reactivos.

6.6. Impacto cultural

El proyecto también tiene un componente cultural importante, al potenciar la cultura de colaboración interdisciplinar entre informática, biología y farmacia. Esta convergencia refuerza la visión de la ciencia como una actividad integradora, capaz de ofrecer soluciones reales a problemas complejos mediante el trabajo conjunto entre perfiles diversos. Además, demuestra el valor de la IA para responder a problemas biomédicos, contribuyendo a difundir la relevancia de la ciencia de datos en la sociedad.

Además, al ser un trabajo académico basado en datos reales, desarrollado dentro de una plataforma abierta como DISNET, contribuye a la difusión del conocimiento y al fomento de la cultura científica. La publicación del código, junto con su documentación, facilita la reutilización del trabajo por parte de otros estudiantes e investigadores, favoreciendo una cultura de ciencia abierta y reproducible.

No obstante, la interpretación incorrecta de los resultados por parte de públicos no especializados podría reforzar la percepción de la IA como una “caja negra” incuestionable, restando valor al pensamiento crítico.

6.7. Contribución a los Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030



Figura 22. Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030. Fuente: [64]

El metamodelo de reposicionamiento de fármacos desarrollado en este TFG se alinea con varios Objetivos de Desarrollo Sostenible, al integrar tecnología avanzada con un enfoque de salud global. A continuación, se detalla su relación con los ODS más relevantes y las decisiones técnicas que respaldan dicha contribución.

ODS3 - Salud y Bienestar: El metamodelo facilita la identificación de nuevas indicaciones terapéuticas a partir de fármacos ya aprobados, reduciendo el tiempo y el coste necesarios para que un tratamiento llegue a los pacientes. Para evaluar su rendimiento se utilizó como métrica principal el f1-score, que equilibra precisión y *recall* y, por tanto, permite minimizar los FN sin aumentar demasiado los FP. Esta elección metodológica resalta el compromiso con la salud pública: se busca no descartar candidatos potencialmente beneficiosos, pero manteniendo el rigor necesario para evitar predicciones erróneas o poco fundamentadas. El enfoque resulta especialmente valioso en enfermedades que carecen de tratamientos específicos, donde cada hipótesis fiable puede suponer un avance significativo.

ODS 9 - Industria, Innovación e Infraestructura: Este proyecto contribuye al ODS 9 al aplicar técnicas avanzadas de ciencia de datos y ML para optimizar las fases tempranas del desarrollo farmacéutico. La priorización de pares enfermedad-fármaco con alta probabilidad de éxito permite a la industria centrar sus recursos en hipótesis más prometedoras, reduciendo costes y tiempos asociados a los procesos experimentales iniciales. De este modo, el metamodelo se plantea como una herramienta estratégica que apoya la innovación industrial en el descubrimiento de nuevos usos terapéuticos para fármacos existentes. Además, el uso de librerías de código abierto (Pandas, Dask, scikit-learn) y datos biomédicos reales refuerza su aplicabilidad en entornos de I+D.

ODS 10 - Reducción de las Desigualdades: El DR resulta especialmente útil para enfermedades raras o desatendidas, ámbitos donde el retorno económico suele ser insuficiente para movilizar inversión privada. De esta forma se

promueve el acceso equitativo a tratamientos innovadores sin depender de grandes presupuestos de investigación.

La disminución de los costes de los nuevos tratamientos por DR también contribuye a reducir las desigualdades en salud entre los distintos países, ya que aumenta las posibilidades de que los países con menos recursos puedan acceder a dichos fármacos más económicos.

ODS 12 - Producción y Consumo Responsables: El enfoque de DR promueve un uso más eficiente de los recursos disponibles, ya que reutiliza compuestos previamente aprobados con un historial de seguridad validado. Esto permite evitar fases iniciales de síntesis, ensayos preclínicos y pruebas en animales, reduciendo el consumo de materiales, reactivos y energía en laboratorios. En este sentido, el metamodelo desarrollado contribuye a priorizar hipótesis más viables, optimizando los procesos de investigación biomédica y reduciendo los residuos generados en las primeras etapas del desarrollo de un nuevo tratamiento.

ODS 17 –Alianzas para Lograr los Objetivos: El proyecto se apoya en la plataforma DISNET, desarrollada en el Medical Data Analytics Laboratory (MEDAL) del Centro de Tecnología Biomédica (CTB), perteneciente a la UPM, y promueve la ciencia abierta al exponer públicamente la memoria metodológica, los resultados principales y el código completo desarrollado. Esta documentación permite que otros investigadores puedan comprender, reproducir y extender el enfoque propuesto. Esta transparencia sienta las bases para nuevas colaboraciones académico - industriales y refuerza la capacidad de compartir conocimiento de forma responsable.

En resumen, el metamodelo no solo aporta valor científico y técnico, sino que se integra en un marco de sostenibilidad y responsabilidad social, contribuyendo de manera tangible a los retos globales planteados en la Agenda 2030.

7 Bibliografía

- [1] K. Park, «A review of computational drug repurposing», *Transl. Clin. Pharmacol.*, vol. 27, n.º 2, p. 59, 2019, doi: 10.12793/tcp.2019.27.2.59.
- [2] F. Saldivar-González, F. D. Prieto-Martínez, J. L. Medina-Franco, F. Saldivar-González, F. D. Prieto-Martínez, y J. L. Medina-Franco, «Descubrimiento y desarrollo de fármacos: un enfoque computacional», *Educ. Quím.*, vol. 28, n.º 1, pp. 51-58, 2017, doi: 10.1016/j.eq.2016.06.002.
- [3] L. P. Santamaría, E. U. Carro, M. D. Uzquiano, E. M. Ruiz, Y. P. Gallardo, y A. Rodríguez-González, «A data-driven methodology towards evaluating the potential of drug repurposing hypotheses», *Comput. Struct. Biotechnol. J.*, vol. 19, pp. 4559-4573, ene. 2021, doi: 10.1016/j.csbj.2021.08.003.
- [4] S. Rodriguez *et al.*, «Machine learning identifies candidates for drug repurposing in Alzheimer's disease», *Nat. Commun.*, vol. 12, n.º 1, p. 1033, feb. 2021, doi: 10.1038/s41467-021-21330-0.
- [5] L. Prieto Santamaría, «Creation, integration, and analysis of disease networks towards a better disease understanding and drug repurposing», PhD Thesis, Universidad Politécnica de Madrid, 2023. doi: 10.20868/UPM.thesis.76526.
- [6] S. Pushpakom *et al.*, «Drug repurposing: progress, challenges and recommendations», *Nat. Rev. Drug Discov.*, vol. 18, n.º 1, pp. 41-58, ene. 2019, doi: 10.1038/nrd.2018.168.
- [7] G. Lagunes-García, A. Rodríguez-González, L. Prieto-Santamaría, E. P. G. del Valle, M. Zanin, y E. Menasalvas-Ruiz, «DISNET: a framework for extracting phenotypic disease information from public sources», *PeerJ*, vol. 8, p. e8580, feb. 2020, doi: 10.7717/peerj.8580.
- [8] M. J. Keiser *et al.*, «Predicting new molecular targets for known drugs», *Nature*, vol. 462, n.º 7270, pp. 175-181, nov. 2009, doi: 10.1038/nature08506.
- [9] G. Wei *et al.*, «Gene expression-based chemical genomics identifies rapamycin as a modulator of MCL1 and glucocorticoid resistance», *Cancer Cell*, vol. 10, n.º 4, pp. 331-342, oct. 2006, doi: 10.1016/j.ccr.2006.09.006.
- [10] J. T. Dudley, T. Deshpande, y A. J. Butte, «Exploiting drug-disease relationships for computational drug repositioning», *Brief. Bioinform.*, vol. 12, n.º 4, pp. 303-311, jul. 2011, doi: 10.1093/bib/bbr013.
- [11] D. B. Kitchen, H. Decornez, J. R. Furr, y J. Bajorath, «Docking and scoring in virtual screening for drug discovery: methods and applications», *Nat. Rev. Drug Discov.*, vol. 3, n.º 11, pp. 935-949, nov. 2004, doi: 10.1038/nrd1549.
- [12] P. Sanseau *et al.*, «Use of genome-wide association studies for drug repositioning», *Nat. Biotechnol.*, vol. 30, n.º 4, pp. 317-320, abr. 2012, doi: 10.1038/nbt.2151.

- [13] C. S. Greene y B. F. Voight, «Pathway and network-based strategies to translate genetic discoveries into effective therapies», *Hum. Mol. Genet.*, vol. 25, n.º R2, pp. R94-R98, oct. 2016, doi: 10.1093/hmg/ddw160.
- [14] S. B. Smith, W. Dampier, A. Tozeren, J. R. Brown, y M. Magid-Slav, «Identification of Common Biological Pathways and Drug Targets Across Multiple Respiratory Viruses Based on Human Host Gene Expression Analysis», *PLOS ONE*, vol. 7, n.º 3, p. e33174, mar. 2012, doi: 10.1371/journal.pone.0033174.
- [15] H. Paik *et al.*, «Repurpose terbutaline sulfate for amyotrophic lateral sclerosis using electronic medical records», *Sci. Rep.*, vol. 5, n.º 1, p. 8580, mar. 2015, doi: 10.1038/srep08580.
- [16] A. Rajput *et al.*, «Prediction of repurposed drugs for Coronaviruses using artificial intelligence and machine learning», *Comput. Struct. Biotechnol. J.*, vol. 19, pp. 3133-3148, 2021, doi: 10.1016/j.csbj.2021.05.037.
- [17] S. Herráiz-Gil, E. Nygren-Jiménez, D. N. Acosta-Alonso, C. León, y S. Guerrero-Aspizua, «Artificial Intelligence-Based Methods for Drug Repurposing and Development in Cancer», *Appl. Sci.*, vol. 15, n.º 5, p. 2798, mar. 2025, doi: 10.3390/app15052798.
- [18] N. T. Issa, V. Stathias, S. Schürer, y S. Dakshanamurthy, «Machine and deep learning approaches for cancer drug repurposing», *Semin. Cancer Biol.*, vol. 68, pp. 132-142, ene. 2021, doi: 10.1016/j.semcancer.2019.12.011.
- [19] A. Ayuso-Muñoz, L. Prieto-Santamaría, E. Ugarte-Carro, E. Serrano, y A. Rodríguez-González, «Uncovering hidden therapeutic indications through drug repurposing with graph neural networks and heterogeneous data», *Artif. Intell. Med.*, vol. 145, p. 102687, nov. 2023, doi: 10.1016/j.artmed.2023.102687.
- [20] T. N. Kipf y M. Welling, «Semi-Supervised Classification with Graph Convolutional Networks», 22 de febrero de 2017, *arXiv*: arXiv:1609.02907. doi: 10.48550/arXiv.1609.02907.
- [21] L. Prieto Santamaría, M. Díaz Uzquiano, E. Ugarte Carro, N. Ortiz-Roldán, Y. Pérez Gallardo, y A. Rodríguez-González, «Integrating heterogeneous data to facilitate COVID-19 drug repurposing», *Drug Discov. Today*, vol. 27, n.º 2, pp. 558-566, feb. 2022, doi: 10.1016/j.drudis.2021.10.002.
- [22] E. Guney, J. Menche, M. Vidal, y A.-L. Barábasi, «Network-based in silico drug efficacy screening», *Nat. Commun.*, vol. 7, n.º 1, p. 10331, feb. 2016, doi: 10.1038/ncomms10331.
- [23] B. Otero-Carrasco, E. Ugarte Carro, L. Prieto-Santamaría, M. Díaz Uzquiano, J. P. Caraça-Valente Hernández, y A. Rodríguez-González, «Identifying patterns to uncover the importance of biological pathways on known drug repurposing scenarios», *BMC Genomics*, vol. 25, n.º 1, p. 43, ene. 2024, doi: 10.1186/s12864-023-09913-1.
- [24] B. Otero Carrasco, L. Prieto-Santamaría, y A. Rodríguez-González, «Exploring Innovative Approaches to Drug Repurposing: Emphasizing the

- Relevance of Biological Pathways», en *REXPO24 Conference*, REPO4EU, abr. 2024. doi: 10.58647/REXPO.24000039.v1.
- [25] J. Amat Rodrigo, «cienciadedatos.net», ene. 2021, doi: 10.5281/ZENODO.10006330.
 - [26] A. J. Smola y B. Schölkopf, «A tutorial on support vector regression», *Stat. Comput.*, vol. 14, n.º 3, pp. 199-222, ago. 2004, doi: 10.1023/B:STCO.0000035301.49549.88.
 - [27] S. R. Safavian y D. Landgrebe, «A survey of decision tree classifier methodology», *IEEE Trans. Syst. Man Cybern.*, vol. 21, n.º 3, pp. 660-674, may 1991, doi: 10.1109/21.97458.
 - [28] L. Breiman, «Random Forests», *Mach. Learn.*, vol. 45, n.º 1, pp. 5-32, oct. 2001, doi: 10.1023/A:1010933404324.
 - [29] T. Chen y C. Guestrin, «XGBoost: A Scalable Tree Boosting System», en *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ago. 2016, pp. 785-794. doi: 10.1145/2939672.2939785.
 - [30] H. Zhang, «The Optimality of Naive Bayes».
 - [31] P. Cunningham y S. J. Delany, «k-Nearest Neighbour Classifiers: 2nd Edition (with Python examples)», *ACM Comput. Surv.*, vol. 54, n.º 6, pp. 1-25, jul. 2022, doi: 10.1145/3459665.
 - [32] Roheen Qamar y Baqar Ali Zardari, «Artificial Neural Networks: An Overview», *Mesopotamian J. Comput. Sci.*, vol. 2023, pp. 124-133, ago. 2023, doi: 10.58496/MJCSC/2023/015.
 - [33] J. Schmidhuber, «Deep learning in neural networks: An overview», *Neural Netw.*, vol. 61, pp. 85-117, ene. 2015, doi: 10.1016/j.neunet.2014.09.003.
 - [34] J. Heaton, «Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning», *Genet. Program. Evolvable Mach.*, vol. 19, n.º 1, pp. 305-307, jun. 2018, doi: 10.1007/s10710-017-9314-z.
 - [35] D. Morselli Gysi *et al.*, «Network medicine framework for identifying drug-repurposing opportunities for COVID-19», *Proc. Natl. Acad. Sci.*, vol. 118, n.º 19, p. e2025581118, may 2021, doi: 10.1073/pnas.2025581118.
 - [36] C. Liu, A. Pavlenko, M. Interlandi, y B. Haynes, «Data formats in analytical DBMSs: performance trade-offs and future directions», *VLDB J.*, vol. 34, n.º 3, p. 30, may 2025, doi: 10.1007/s00778-025-00911-1.
 - [37] M. Rocklin, «Dask: Parallel Computation with Blocked algorithms and Task Scheduling», presentado en Python in Science Conference, Austin, Texas, 2015, pp. 126-132. doi: 10.25080/Majora-7b98e3ed-013.
 - [38] W. McKinney, «Data Structures for Statistical Computing in Python», presentado en Python in Science Conference, Austin, Texas, 2010, pp. 56-61. doi: 10.25080/Majora-92bf1922-00a.
 - [39] F. Pedregosa *et al.*, «Scikit-learn: Machine Learning in Python», *Mach. Learn. PYTHON*.

- [40] «Comparative Analysis of Data Visualization Libraries Matplotlib and Seaborn in Python», *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 10, n.º 1, pp. 277-281, feb. 2021, doi: 10.30534/ijatcse/2021/391012021.
- [41] «Guía de CRISP-DM de IBM SPSS Modeler».
- [42] J. F. V. Rueda, «CRISP-DM: una metodología para minería de datos en salud», *healthdataminer.com*. Accedido: 26 de mayo de 2025. [En línea]. Disponible en: <https://healthdataminer.com/data-mining/crisp-dm-una-metodologia-para-mineria-de-datos-en-salud/>
- [43] G. A. Pradipta, R. Wardoyo, A. Musdholifah, I. N. H. Sanjaya, y M. Ismail, «SMOTE for Handling Imbalanced Data Problem : A Review», en *2021 Sixth International Conference on Informatics and Computing (ICIC)*, nov. 2021, pp. 1-8. doi: 10.1109/ICIC54025.2021.9632912.
- [44] L. Yu y N. Zhou, «Survey of Imbalanced Data Methodologies», 6 de abril de 2021, *arXiv*: arXiv:2104.02240. doi: 10.48550/arXiv.2104.02240.
- [45] E. Orellana, «SMOTE», *Medium*. Accedido: 3 de junio de 2025. [En línea]. Disponible en: <https://emilia-orellana44.medium.com/smote-2acd5dd09948>
- [46] «¿Qué es el submuestreo? | IBM». Accedido: 3 de junio de 2025. [En línea]. Disponible en: <https://www.ibm.com/es-es/think/topics/downsampling>
- [47] S. Varma y R. Simon, «Bias in error estimation when using cross-validation for model selection», *BMC Bioinformatics*, vol. 7, n.º 1, p. 91, feb. 2006, doi: 10.1186/1471-2105-7-91.
- [48] M. Zitnik, R. Sosič, y J. Leskovec, «Prioritizing network communities», *Nat. Commun.*, vol. 9, n.º 1, p. 2544, jun. 2018, doi: 10.1038/s41467-018-04948-5.
- [49] D. L. Spark, A. Fornito, C. J. Langmead, y G. D. Stewart, «Beyond antipsychotics: a twenty-first century update for preclinical development of schizophrenia therapeutics», *Transl. Psychiatry*, vol. 12, n.º 1, pp. 1-11, abr. 2022, doi: 10.1038/s41398-022-01904-2.
- [50] J. Rivers-Auty, A. E. Mather, R. Peters, C. B. Lawrence, y D. Brough, «Anti-inflammatories in Alzheimer's disease—potential therapy or spurious correlate?», *Brain Commun.*, vol. 2, n.º 2, p. fcaa109, jul. 2020, doi: 10.1093/braincomms/fcaa109.
- [51] M. A. Salem, B. Budzyńska, J. Kowalczyk, N. S. El Sayed, y S. M. Mansour, «Tadalafil and bergapten mitigate streptozotocin-induced sporadic Alzheimer's disease in mice via modulating neuroinflammation, PI3K/Akt, Wnt/ β -catenin, AMPK/mTOR signaling pathways», *Toxicol. Appl. Pharmacol.*, vol. 429, p. 115697, oct. 2021, doi: 10.1016/j.taap.2021.115697.
- [52] M. L. Crismon, «Tacrine: First Drug Approved for Alzheimer's Disease», *Ann. Pharmacother.*, vol. 28, n.º 6, pp. 744-751, jun. 1994, doi: 10.1177/106002809402800612.
- [53] J. T. Olin y L. Schneider, «Galantamine for Alzheimer's disease - Olin, JT - 2002 | Cochrane Library», Accedido: 30 de mayo de 2025. [En línea].

- Disponible en:
<https://www.cochranelibrary.com/cdsr/doi/10.1002/14651858.CD001747/full>
- [54] J. S. Birks y J. G. Evans, «Rivastigmine for Alzheimer's disease - Birks, JS - 2015 | Cochrane Library», Accedido: 30 de mayo de 2025. [En línea]. Disponible en:
<https://www.cochranelibrary.com/cdsr/doi/10.1002/14651858.CD001191.pub3/full>
- [55] Z. Zhang *et al.*, «Melatonin: A potential nighttime guardian against Alzheimer's», *Mol. Psychiatry*, vol. 30, n.º 1, pp. 237-250, ene. 2025, doi: 10.1038/s41380-024-02691-6.
- [56] N. Tabet y H. Feldman, «Ibuprofen for Alzheimer's disease - Tabet, N - 2003 | Cochrane Library», Accedido: 30 de mayo de 2025. [En línea]. Disponible en:
<https://www.cochranelibrary.com/cdsr/doi/10.1002/14651858.CD004031/full>
- [57] R. Howard *et al.*, «Minocycline at 2 Different Dosages vs Placebo for Patients With Mild Alzheimer Disease: A Randomized Clinical Trial», *JAMA Neurol.*, vol. 77, n.º 2, p. 164, feb. 2020, doi: 10.1001/jamaneurol.2019.3762.
- [58] P. A. Newhouse, A. Potter, M. Kelton, y J. Corwin, «Nicotinic treatment of Alzheimer's disease», *Biol. Psychiatry*, vol. 49, n.º 3, pp. 268-278, feb. 2001, doi: 10.1016/S0006-3223(00)01069-6.
- [59] S.-C. Chang, C.-H. Chen, y M.-L. Lu, «Cabergoline-induced psychotic exacerbation in schizophrenic patients», *Gen. Hosp. Psychiatry*, vol. 30, n.º 4, pp. 378-380, jul. 2008, doi: 10.1016/j.genhosppsych.2007.11.002.
- [60] A. Gogos, A. M. Sbisa, J. Sun, A. Gibbons, M. Udawela, y B. Dean, «A Role for Estrogen in Schizophrenia: Clinical and Preclinical Findings», *Int. J. Endocrinol.*, vol. 2015, n.º 1, p. 615356, 2015, doi: 10.1155/2015/615356.
- [61] B. Alevizos, «Dependence and chronic psychosis with D-nor-pseudoephedrine», *Eur. Psychiatry*, vol. 18, n.º 8, pp. 423-425, dic. 2003, doi: 10.1016/j.eurpsy.2003.05.001.
- [62] P. Sivaraman, R. D. Rattehalli, y M. B. Jayaram, «Levomepromazine for schizophrenia - Sivaraman, P - 2010 | Cochrane Library», Accedido: 30 de mayo de 2025. [En línea]. Disponible en:
<https://www.cochranelibrary.com/cdsr/doi/10.1002/14651858.CD007779.pub2/abstract>
- [63] A. Fitton y R. C. Heel, «Clozapine», *Drugs*, vol. 40, n.º 5, pp. 722-747, nov. 1990, doi: 10.2165/00003495-199040050-00007.
- [64] M. J. Gamez, «Objetivos y metas de desarrollo sostenible», Desarrollo Sostenible. Accedido: 23 de mayo de 2025. [En línea]. Disponible en:
<https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>

Anexos

Índice de Tablas de los Anexos

Tabla del Anexo I. Combinaciones de hiperparámetros para cada modelo.	61
Tabla del Anexo II. Resultados por <i>pipeline</i> y método en validación cruzada..	64

Índice de Figuras de los Anexos

Figura del Anexo I. Histogramas de las métricas de GNN.	62
Figura del Anexo II. Histogramas de las métricas de <i>Information Paths</i>	62
Figura del Anexo III. Histogramas de las métricas de <i>Network Proximity</i>	63

Anexo I: Configuración de hiperparámetros

Para cada modelo se exploraron las siguientes combinaciones de hiperparámetros dentro del procedimiento *GridSearchCV*. Las combinaciones óptimas por modelo y *pipeline* están disponibles en GitLab.

MODELO	PARÁMETRO	VALORES EVALUADOS
KNN <i>KNeighborsClassifier</i>	<i>n_neighbors</i>	[3, 5, 7, 9]
	<i>weights</i>	["uniform", "distance"]
	<i>metric</i>	["euclidean", "manhattan"]
Random Forest <i>RandomForestClassifier</i>	<i>n_estimators</i>	[100, 200]
	<i>max_depth</i>	[None, 10, 20]
	<i>min_samples_split</i>	[2, 5]
	<i>Class_weight</i>	["balanced"]
Decision Tree <i>DecisionTreeClassifier</i>	<i>max_depth</i>	[None, 10, 20]
	<i>min_samples_split</i>	[2, 5]
	<i>criterion</i>	["gini", "entropy", "log_loss"]
	<i>Class_weight</i>	["balanced"]
Logistic Regression <i>LogisticRegression</i>	<i>C</i>	[0.01, 0.1, 1, 10, 100]
	<i>solver</i>	["liblinear", "saga"]
	<i>penalty</i>	["l1", "l2"]
	<i>Class_weight</i>	["balanced"]
XGBoost <i>XGBClassifier</i> (<i>eval_metric</i> ="logloss")	<i>n_estimators</i>	[10, 50, 100, 150, 500]
	<i>max_depth</i>	[2, 3, 5, 7, 10]
	<i>learning_rate</i>	[0.01, 0.1, 0.2]
	<i>Scale_pos_weight</i>	(<i>y_train</i> == 0).sum() / (<i>y_train</i> == 1).sum())

Tabla del Anexo I. Combinaciones de hiperparámetros para cada modelo.

Anexo II: Distribución de las métricas

En este anexo se incluyen los histogramas (en escala logarítmica) correspondientes a cada variable numérica de la tabla final integrada. Estas visualizaciones permiten identificar la asimetría de las métricas derivadas de GNN (concentradas en valores próximos a 1), la dispersión y varianza de las métricas de *Network Proximity*, y la baja frecuencia de activación de algunos *paths*, complementado el análisis descriptivo realizado durante el EDA.

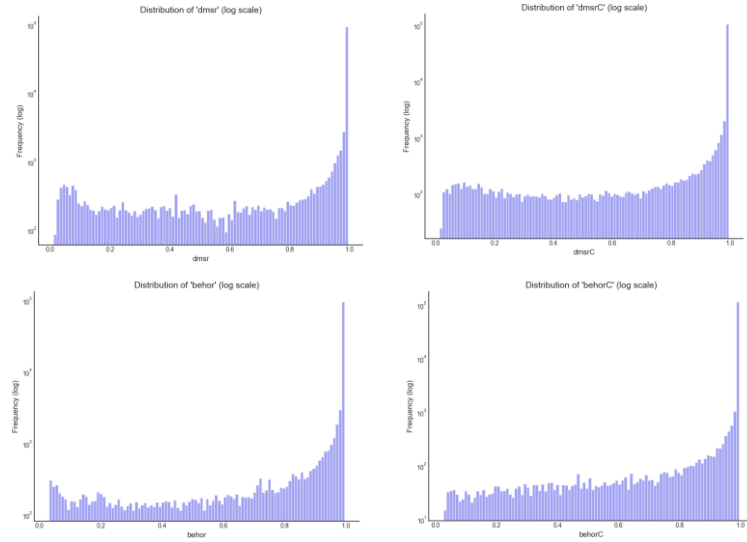


Figura del Anexo I. Histogramas de las métricas de GNN.

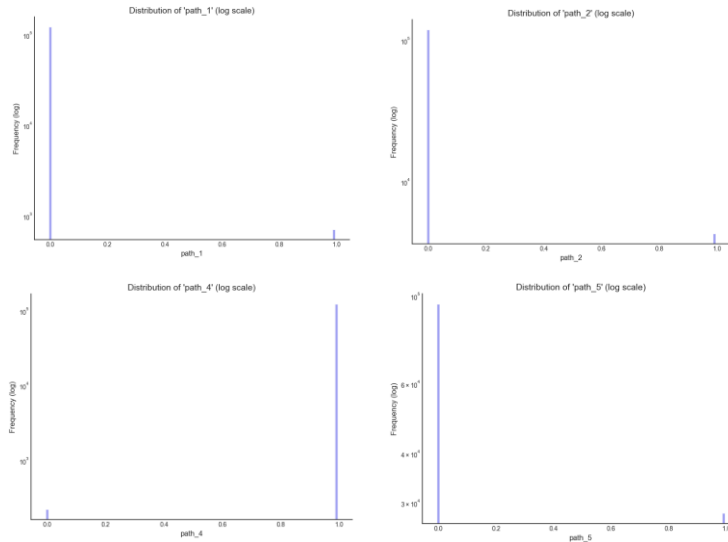


Figura del Anexo II. Histogramas de las métricas de *Information Paths*.

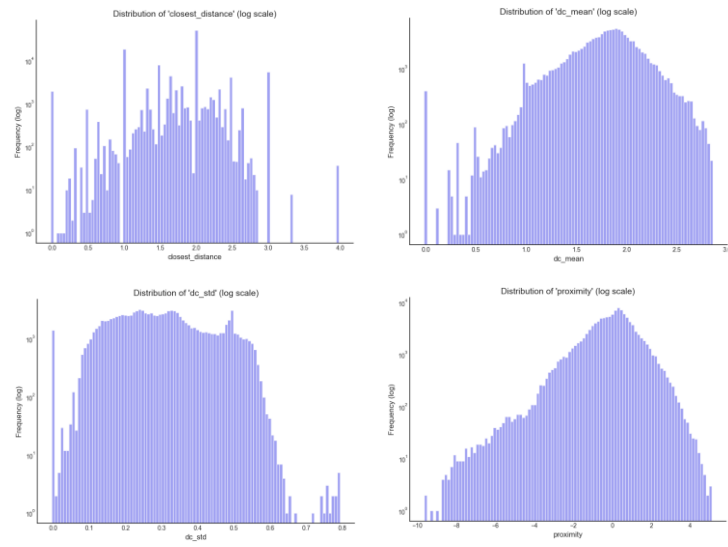


Figura del Anexo III. Histogramas de las métricas de *Network Proximity*.

Anexo III: Resultados completos de la validación cruzada

En este anexo se muestran en una tabla los resultados detallados obtenidos durante la validación cruzada anidada para cada combinación de *pipeline* y método de clasificación. Se incluyen las medias de las métricas principales calculadas sobre los cinco *folds*, junto con su desviación estándar para cada una de ellas.

Pipeline	Modelo	F1 class1	F1 class0	Precision	Recall	Auc Roc	TP	TN	FP	FN
Cost Sensitive	Decision Tree	0.269 ± 0.013	0.973 ± 0.002	0.278 ± 0.022	0.263 ± 0.022	0.625 ± 0.013	223.8 ± 18.807	22161.4 ± 92.546	587.4 ± 92.343	625.6 ± 19.1
	Logistic Regresion	0.191 ± 0.008	0.956 ± 0.005	0.149 ± 0.012	0.272 ± 0.032	0.772 ± 0.008	231.2 ± 26.948	21411.2 ± 255.483	1337.6 ± 255.747	618.2 ± 27.234
	Random Forest	0.374 ± 0.014	0.973 ± 0.002	0.33 ± 0.029	0.435 ± 0.039	0.863 ± 0.006	369.2 ± 32.995	21990.4 ± 134.443	758.4 ± 134.415	480.2 ± 33.056
	XGBoost	0.33 ± 0.013	0.971 ± 0.004	0.293 ± 0.027	0.385 ± 0.05	0.879 ± 0.005	326.6 ± 42.84	21944.2 ± 202.78	804.6 ± 202.515	522.8 ± 42.892
Downsampling	Decision Tree	0.862 ± 0.006	0.832 ± 0.009	0.791 ± 0.01	0.948 ± 0.009	0.879 ± 0.006	805.2 ± 7.759	636.4 ± 13.795	213.0 ± 13.491	44.2 ± 7.497
	KNN	0.82 ± 0.007	0.739 ± 0.011	0.71 ± 0.007	0.968 ± 0.008	0.828 ± 0.006	822.4 ± 7.162	514.2 ± 10.756	335.2 ± 10.941	27.0 ± 6.892
	Logistic Regresion	0.836 ± 0.008	0.774 ± 0.015	0.735 ± 0.011	0.971 ± 0.003	0.846 ± 0.013	824.4 ± 2.408	552.0 ± 16.583	297.4 ± 16.41	25.0 ± 2.449
	Random Forest	0.879 ± 0.007	0.86 ± 0.011	0.826 ± 0.017	0.939 ± 0.015	0.926 ± 0.009	797.4 ± 13.012	680.8 ± 22.095	168.6 ± 22.379	52.0 ± 13.115
	XGBoost	0.876 ± 0.005	0.858 ± 0.004	0.824 ± 0.006	0.935 ± 0.016	0.923 ± 0.008	794.4 ± 13.993	679.6 ± 10.526	169.8 ± 10.06	55.0 ± 13.784
No Sampling	Decision Tree	0.272 ± 0.016	0.973 ± 0	0.273 ± 0.012	0.271 ± 0.022	0.623 ± 0.01	230.4 ± 18.366	22135.2 ± 30.111	613.6 ± 29.796	619.0 ± 18.466
	KNN	0.283 ± 0.019	0.973 ± 0.002	0.289 ± 0.026	0.282 ± 0.036	0.693 ± 0.012	239.2 ± 30.548	22153.0 ± 119.912	595.8 ± 119.857	610.2 ± 30.54
	Logistic Regresion	0.191 ± 0.008	0.955 ± 0.01	0.153 ± 0.033	0.275 ± 0.052	0.772 ± 0.006	233.4 ± 43.901	21374.0 ± 465.18	1374.8 ± 465.594	616.0 ± 44.164
	Random Forest	0.376 ± 0.018	0.973 ± 0.002	0.336 ± 0.024	0.429 ± 0.022	0.852 ± 0.003	364.0 ± 18.317	22027.2 ± 75.658	721.6 ± 75.411	485.4 ± 18.501
	XGBoost	0.342 ± 0.009	0.974 ± 0.002	0.326 ± 0.021	0.361 ± 0.021	0.884 ± 0.004	306.6 ± 17.672	22108.6 ± 91.719	640.2 ± 92.053	542.8 ± 17.852
Oversampling	Decision Tree	0.926 ± 0.002	0.925 ± 0.002	0.918 ± 0.002	0.935 ± 0.002	0.926 ± 0.002	21261.2 ± 56.309	20837.6 ± 57.886	1911.2 ± 57.647	1487.6 ± 56.208
	KNN	0.908 ± 0.001	0.898 ± 0.002	0.866 ± 0.004	0.955 ± 0.004	0.944 ± 0.001	21714.0 ± 85.484	19374.6 ± 126.792	3374.2 ± 126.581	1034.8 ± 85.555
	Logistic Regresion	0.736 ± 0.002	0.569 ± 0.004	0.616 ± 0.002	0.913 ± 0.003	0.776 ± 0.004	20758.4 ± 58.731	9829.6 ± 109.111	12919.2 ± 108.955	1990.4 ± 58.607
	Random Forest	0.949 ± 0.001	0.948 ± 0.001	0.935 ± 0.003	0.964 ± 0.005	0.985 ± 0.001	21939.6 ± 121.434	21223.2 ± 86.537	1525.6 ± 86.892	809.2 ± 121.714
	XGBoost	0.94 ± 0.002	0.937 ± 0.002	0.917 ± 0.004	0.965 ± 0.002	0.983 ± 0.001	21942.4 ± 44.892	20757.8 ± 109.561	1991.0 ± 109.186	806.4 ± 45.258
Oversampling Gridsearch	Decision Tree	0.26 ± 0.004	0.95 ± 0.002	0.182 ± 0.004	0.453 ± 0.014	0.699 ± 0.01	384.8 ± 12.153	21022.8 ± 81.641	1726.0 ± 81.649	464.6 ± 12.157
	KNN	0.247 ± 0.009	0.956 ± 0.004	0.185 ± 0.014	0.375 ± 0.031	0.74 ± 0.017	318.6 ± 26.52	21329.6 ± 198.51	1419.2 ± 198.405	530.8 ± 26.527
	Logistic Regresion	0.191 ± 0.007	0.957 ± 0.006	0.151 ± 0.014	0.267 ± 0.034	0.774 ± 0.009	226.8 ± 29.115	21451.2 ± 293.252	1297.6 ± 293.515	622.6 ± 29.365
	Random Forest	0.325 ± 0.005	0.967 ± 0.002	0.264 ± 0.009	0.423 ± 0.018	0.86 ± 0.004	359.2 ± 15.007	21747.2 ± 85.186	1001.6 ± 85.483	490.2 ± 15.205
	XGBoost	0.346 ± 0.013	0.974 ± 0.004	0.338 ± 0.054	0.369 ± 0.053	0.879 ± 0.004	313.4 ± 44.562	22101.2 ± 220.791	647.6 ± 221.156	536.0 ± 44.76

Tabla del Anexo II. Resultados por *pipeline* y método en validación cruzada.

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=ETS Ingenieros Informaticos - UPM, C=ES
Fecha/Hora	Wed Jun 04 09:48:06 CEST 2025
Emisor del Certificado	EMAILADDRESS=camanager@etsiinf.upm.es, CN=CA ETS Ingenieros Informaticos, O=ETS Ingenieros Informaticos - UPM, C=ES
Numero de Serie	561
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)