



Trabajo de Fin de Grado

Machine Learning como alternativa a los modelos de simulación para la toma de decisiones

*Machine Learning as an alternative to simulation models
for decision making*

Marco Antonio Cabrera Hernández

La Laguna, 22 de mayo de 2023

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

D. **Iván Castilla Rodríguez**, con NIF 78.565.451-G profesor Contratado doctor adscrito al Departamento de Nombre del Departamento de la Universidad de La Laguna, como tutor.

D. **Rafael Arnay del Arco**, con NIF 78.569.591-G profesor Contratado doctor adscrito al Departamento de Nombre del Departamento de la Universidad de La Laguna, como cotutor.

CERTIFICA(N)

Que la presente memoria titulada:

“Machine Learning como alternativa a los modelos de simulación para la toma de decisiones” ha sido realizada bajo su dirección por D. **Iván Castilla Rodríguez**, con NIF 78.565.451-G.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna, en la fecha que figura en la firma electrónica.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Arnay del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

Agradecimientos

A mi familia,

gracias por estar siempre a mi lado, por el apoyo incondicional y por ayudarme a alcanzar todas mis metas. Sin ustedes, este logro no sería posible.

A ti, Mire,

gracias por tus palabras de motivación, tu confianza constante y tu creencia en mí incluso cuando dudaba de mis propias habilidades. No hay palabras suficientes para expresar lo agradecido que estoy por tenerte en mi vida.

A Iván y Rafa,

agradezco su dedicación y compromiso en guiarme a través de este proceso. Sus comentarios, correcciones y valiosas sugerencias han enriquecido mi trabajo y me han ayudado a crecer como profesional. Gracias por compartir su tiempo, sabiduría y experiencia conmigo.

A todos aquellos que han sido parte de este viaje,

a mis amigos, compañeros de clase y demás seres queridos que me han brindado su apoyo incondicional y comprensión a lo largo de este tiempo, gracias por estar ahí en cada paso del camino. Sus palabras de aliento, su motivación constante y su amistad sincera han sido un impulso invaluable. Agradezco su presencia en mi vida y por ser una parte importante de esta historia.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Arnay del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

Licencia



© Esta obra está bajo una licencia de Creative
CommonsReconocimiento-CompartirIgual 4.0 Internacional.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Arnay del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

Resumen

Existen multitud de campos de aplicación donde es habitual el uso de la simulación para predecir el comportamiento a largo plazo del sistema modelado. Según aumenta la complejidad de estos modelos, aumenta también el tiempo de cómputo necesario para obtener resultados. Una alternativa es crear un modelo con técnicas de aprendizaje automático que imite el comportamiento del modelo original, pero cuyo tiempo de cómputo es casi instantáneo: estos son llamados modelos subrogados. En este proyecto se ha realizado una comparativa entre un modelo de simulación para evaluar la efectividad, seguridad y coste-efectividad de los sistemas de monitorización continua de glucosa intersticial en tiempo real para diabetes tipo 1 y modelos subrogados de aprendizaje automático, analizando métricas de rendimiento como el coeficiente de determinación y el error porcentual absoluto medio. Los resultados obtenidos han demostrado que los modelos subrogados son capaces de capturar de manera precisa y eficiente el comportamiento del sistema modelado en un tiempo significativamente menor; lo que los convierte en una herramienta valiosa para su integración con herramientas para la toma de decisiones en tiempo real.

Palabras clave: modelo de simulación, modelos subrogados, diabetes.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015. Su autenticidad puede ser contrastada en la siguiente dirección https://sede.ull.es/validacion/	
Identificador del documento: 5429785	Código de verificación: nPGy9YjY
Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

Abstract

There are many fields of application where it is common to use simulation to predict the long-term behavior of the modeled system. As the complexity of these models increases, so does the computational time required to obtain results. An alternative is to create a model with machine learning techniques that mimics the behavior of the original model, but whose computation time is almost instantaneous: these are called surrogate models. In this project, a comparison has been made between a simulation model to evaluate the effectiveness, safety and cost-effectiveness of real-time continuous interstitial glucose monitoring systems for type 1 diabetes and surrogate machine learning models, analyzing performance metrics such as the coefficient of determination and the mean absolute percentage error. The results obtained have shown that surrogate models are able to accurately and efficiently capture the behavior of the modeled system in significantly less time, making them a valuable tool for real-time decision making applications.

Keywords: *simulation model, surrogate models, diabetes.*

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

1. Introducción	8
1.1. Motivación	8
1.2. Objetivos	9
1.3. Antecedentes y estado del arte	9
1.3.1. Diabetes	9
1.3.2. Modelo de simulación	10
1.3.3. Modelos subrogados	10
1.3.4. Aplicación DIAB+	11
1.4. Fases del proyecto	11
2. Materiales y método	13
2.1. Conjunto de datos	14
2.2. Hardware y software	16
2.2.1. Hardware	16
2.2.2. Software	16
2.3. Procedimiento de entrenamiento y evaluación de modelos	18
2.3.1. Entrenamiento de los modelos	19
2.3.1.1. Experimentos realizados	20
2.3.1.2. Tipos de entrenamiento	21
2.3.2. Evaluación de los modelos	22
2.3.3. Algoritmos utilizados	23
2.3.4. Métricas utilizadas	24
2.4. Conexión de los modelos con la aplicación	25
3. Resultados	27
3.1. Elección de las métricas de evaluación	27
3.2. Tipos de entrenamiento	28
3.2.1. Experimento sin manifestaciones iniciales	28
3.2.2. Experimento con manifestaciones iniciales	29
3.3. Ajuste de los modelos	31
3.4. Comparación de los algoritmos utilizados	32
3.5. Análisis de los modelos subrogados	34
3.5.1. Sobreestimación y difícil ajuste en el grupo de enfermedades cardíacas	34
3.5.2. Implicaciones de las manifestaciones iniciales	36
3.5.3. Aceleración en los tiempos de cómputo	38
3.6. Conexión de los modelos con la aplicación	38
4. Discusión	41
4.1. Elección del coeficiente de determinación	41
4.2. Ajuste de los modelos	42
4.3. Sobreestimación de enfermedades cardíacas	43
5. Conclusiones	45
6. Conclusions	46
7. Presupuesto	47

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Capítulo 1 -

1. Introducción

Este capítulo incluye información sobre la motivación y los objetivos del proyecto, así como un análisis del estado del arte respecto a la utilización de modelos de simulación y de aprendizaje automático en el contexto de la diabetes mellitus tipo 1 (DM1). Además, se proporciona una breve descripción sobre ella y los modelos de aprendizaje automático subrogados, también se describe el modelo de simulación existente y su uso en la predicción del curso de la enfermedad. Por último, se describen las fases del proyecto y cómo se llevarán a cabo.

1.1. Motivación

Un modelo de simulación es un modelo matemático que representa un evento o proceso real. Sin embargo, los modelos de simulación pueden ser engorrosos y lentos de ejecutar, especialmente cuando se trata de sistemas complejos o grandes conjuntos de datos.

Aquí es donde entran en juego los modelos de aprendizaje automático subrogado. Estos modelos están diseñados utilizando técnicas de aprendizaje automático para predecir y representar el comportamiento del modelo de simulación. En lugar de ejecutar repetidamente el modelo de simulación para realizar análisis o tomar decisiones, se puede utilizar el modelo subrogado, que es mucho más rápido de evaluar.

Los modelos subrogados se entrenan utilizando datos generados por el modelo de simulación original. Estos datos incluyen diferentes combinaciones de variables de entrada y los resultados correspondientes obtenidos del modelo de simulación. El aprendizaje automático se utiliza para encontrar relaciones y patrones en estos datos y construir un modelo que pueda predecir los resultados del modelo de simulación con precisión aceptable.

Una vez que se ha construido el modelo subrogado, se puede utilizar para realizar

análisis rápidos, explorar diferentes escenarios y tomar decisiones informadas sin tener que ejecutar repetidamente el modelo de simulación. Esto puede conducir a ahorros significativos en tiempo y recursos computacionales.

1.2. Objetivos

En el contexto de este proyecto, se dispone de un modelo de simulación que permite realizar predicciones sobre la evolución de la diabetes en un sujeto, en función de un conjunto de características médicas. Este modelo ha sido implementado por el tutor del proyecto y tiene como objetivo proporcionar información útil para la toma de decisiones y el seguimiento del tratamiento de esta enfermedad.

El objetivo principal es adaptar el modelo de simulación existente a modelos de aprendizaje automático mediante el uso de modelos subrogados, lo que es útil cuando el modelo de simulación es muy lento o costoso de ejecutar, como en este caso.

Para ello, se quiere seleccionar y entrenar distintos modelos de aprendizaje automático utilizando técnicas y algoritmos adecuados para el problema en cuestión. Una vez entrenados, comparar los resultados obtenidos por el modelo de simulación con los obtenidos por los modelos de aprendizaje automático y evaluar si estos últimos son capaces de realizar predicciones de forma más rápida y precisa. Además, se pretende conectarlos a una aplicación web.

1.3. Antecedentes y estado del arte

Esta sección tiene como objetivo proporcionar contexto y orientar al lector sobre el tema que se está abordando.

1.3.1. Diabetes

Según la Organización Mundial de la Salud «la diabetes sacarina o diabetes mellitus es una enfermedad crónica que se presenta cuando el páncreas no secreta suficiente insulina o cuando el organismo no utiliza eficazmente la insulina que produce» [18].

«En 2017, había 9 millones de personas con DM1, la mayoría de las cuales vivía en países de renta alta. Hoy por hoy aún se desconoce la causa de este tipo de diabetes y

tampoco se sabe cómo prevenirla» [18].

Actualmente, existen numerosos modelos de simulación que se encargan de generar escenarios futuros para el desarrollo de algún tipo de enfermedad. En el caso de la diabetes, existe incluso una competición donde diferentes equipos de investigación presentan anualmente modelos para predecir cómo evolucionará un paciente en función de sus características [15].

1.3.2. Modelo de simulación

El modelo de simulación es el descrito en el Informe de Evaluación de Tecnologías Sanitarias (SESCS), publicado en octubre de 2020 [5]. Este informe, describe un modelo matemático que se utiliza para evaluar la efectividad, seguridad y coste-efectividad de los sistemas de monitorización continua de glucosa intersticial en tiempo real (MCG-TR) para DM1.

El modelo tiene en cuenta cuatro grupos de complicaciones crónicas de la diabetes: enfermedad cardiovascular, nefropatía, neuropatía y retinopatía, y también incluye episodios de hipoglucemia graves. El riesgo de progresión de estas complicaciones depende principalmente del nivel de hemoglobina glicosilada (HbA1c) de cada individuo, pero también está influenciado por factores como la edad, el sexo y la duración de la diabetes.

Se contempla, también, la aparición de complicaciones crónicas como eventos que modifican el estado del paciente y pueden aumentar el riesgo de otras complicaciones o aumentar la mortalidad. Además, se tiene en cuenta el coste de tratamiento y seguimiento de las complicaciones crónicas y la calidad de vida asociada a ellas.

1.3.3. Modelos subrogados

Los modelos subrogados son herramientas que se utilizan para sustituir a modelos complejos o costosos en términos de tiempo y recursos, con el fin de obtener resultados rápidos y precisos. Estos modelos son especialmente útiles en campos como la informática y las matemáticas, donde se requieren cálculos y simulaciones complejas.

En el ámbito médico, se han estado utilizando modelos subrogados en los últimos años

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

para reducir el costo computacional y simplificar su complejidad. Estos modelos se usan, por ejemplo, para predecir las salidas de modelos más complejos en la detección de cáncer en imágenes médicas a través de la optimización de parámetros en algoritmos evolutivos [27].

Además, estos modelos son útiles para explicar las decisiones de modelos menos interpretables y complejos en diferentes áreas de la medicina, como la diabetes y la predicción del riesgo de enfermedades cardiovasculares [14] [6]. Su objetivo es mejorar la capacidad explicativa de los diagnósticos y hacer que la toma de decisiones clínicas sea más efectiva.

1.3.4. Aplicación DIAB+

Uno de los objetivos de este proyecto, como se explicó en el apartado 1.2, es conectar los modelos subrogados con una aplicación web. DIAB+ ha sido implementada para generar simulaciones utilizando el modelo descrito en el apartado anterior, el resultado es una serie de gráficas que muestran los resultados de esa misma simulación [2].

Surgió la iniciativa de que la aplicación fuera más interactiva para brindar a los usuarios una experiencia fluida y en tiempo real. Sin embargo, se ha enfrentado a un desafío importante debido a la lentitud en la ejecución del modelo de simulación.

Para superar esta limitación se ha pretendido integrar los modelos subrogados en la aplicación web para reducir significativamente los tiempos de espera al realizar simulaciones y generar visualizaciones de los resultados.

1.4. Fases del proyecto

El proyecto se divide en las siguientes fases:

1. Familiarización con el tema propuesto: se llevará a cabo una revisión bibliográfica sobre el aprendizaje automático y se leerá un libro específico sobre esta temática para obtener una comprensión general de los conceptos y técnicas involucradas [8].
2. Comprensión del modelo de simulación: se examinará el modelo de simulación

proporcionado por el tutor y se entenderá su funcionamiento y propósito.

3. Trabajo con los datos del modelo de simulación: se explorarán los datos proporcionados por el modelo de simulación para obtener una idea de sus ventajas e inconvenientes y para determinar qué características son importantes para predecir la evolución de la diabetes.
4. Preparación de los datos para la utilización de modelos subrogados de aprendizaje automático: se prepararán los datos para ser utilizados por los algoritmos de aprendizaje automático.
5. Selección y entrenamiento de modelos de regresión: se seleccionará un conjunto de modelos de regresión y se entrenarán para predecir la evolución de diferentes enfermedades en pacientes con DM1.
6. Ajuste de los modelos: una vez entrenados, se ajustarán los modelos para mejorar su precisión y minimizar el error.
7. Comparación de resultados: se compararán los resultados obtenidos por los modelos de aprendizaje automático con los del modelo de simulación y se extraerán conclusiones acerca de su rendimiento y eficiencia en relación con el modelo de simulación.
8. Conexión de los modelos con la aplicación web: se conectarán los modelos que mejores resultados proporcionen con la aplicación para suplir el modelo de simulación actual.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Capítulo 2 -

2. Materiales y método

En esta sección se describe el conjunto de datos y se proporciona información sobre el hardware y el software utilizado en el desarrollo del proyecto. Se detalla el procedimiento de entrenamiento y evaluación de los modelos de aprendizaje automático y se describen las métricas utilizadas para evaluar el rendimiento de estos modelos. La Figura 2.1 muestra una idea general del método empleado para el proceso de modelado de los modelos subrogados de aprendizaje automático.

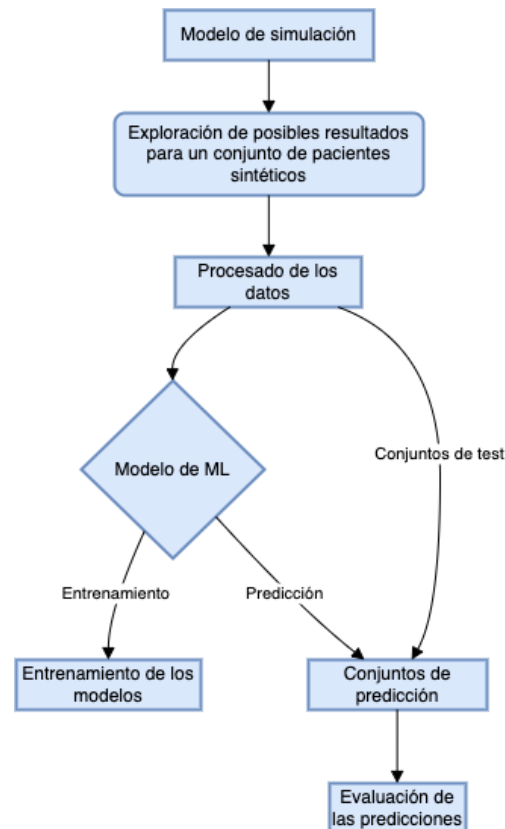


Figura 2.1. Diagrama de flujo del proceso de modelado de aprendizaje automático subrogados.

2.1. Conjunto de datos

En este apartado se explican los datos utilizados, las variables que se utilizan como entrada para los modelos y las variables de salida, junto con sus acrónimos en el archivo Excel donde se encuentran alojados.

El conjunto de datos contiene los resultados de simular diferentes réplicas de pacientes con DM1. Para cada réplica, se realizan 200 simulaciones variando los parámetros del modelo de simulación teniendo en cuenta la incertidumbre que se tiene sobre el valor de esos parámetros; y en cada réplica se simula a 500 pacientes con las mismas características iniciales, para tener en cuenta la incertidumbre estocástica sobre la evolución de estos pacientes.

Es importante mencionar que, aunque los datos generados por un modelo de simulación pueden ser útiles para predecir resultados y evaluar escenarios hipotéticos, hay que tener en cuenta que los resultados obtenidos no siempre son idénticos a los resultados que se obtendrían con datos reales.

Los datos de entrada para los modelos incluyen la edad de los pacientes (AGE), que varía entre 20 y 50 años, la duración de la diabetes (DURATION), que varía de 5 a 30 años, ambos fragmentados en intervalos de 5 años, también se incluyen datos sobre el nivel de HbA1c (HBA1C) y el sexo de cada paciente (SEX).

Además, incluye la tasa anual de episodios graves de hipoglucemia (HYPO_RATE), que se divide en tres niveles: 0, 2 y 4; y se han registrado diferentes configuraciones de manifestaciones iniciales, incluyendo aquellas sin manifestaciones, y aquellas con 1, 2 o 3 manifestaciones al azar para cada configuración de los otros parámetros. En concreto, éstas abarcan el siguiente conjunto de comorbilidades: amputación de extremidades inferiores (LEA), angina (ANGINA), ceguera (BLI), enfermedad renal terminal (ESRD), edema macular diabético (ME), fallo cardíaco (HF), ictus (STROKE), infarto de miocardio (MI), macroalbuminuria (ALB2), microalbuminuria (ALB1), neuropatía (NEU), retinopatía de fondo (BGRET) y retinopatía proliferativa (PRET). La Figura 2.2 muestra este conjunto y sus agrupaciones.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

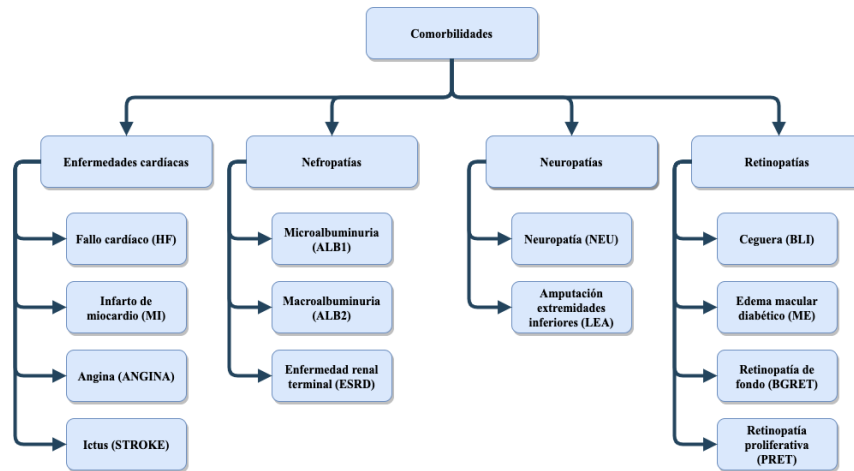


Figura 2.2. Diagrama de agrupaciones de comorbilidades y respectivas abreviaturas.

Para los parámetros mencionados en el párrafo anterior, se ha optado por realizar una exploración aleatoria. Esta elección se ha realizado con el objetivo de evitar llevar a cabo una inabordable exploración de todas las posibles combinaciones de experimentos, mientras que para el resto de parámetros, se ha llevado a cabo una exploración sistemática.

Como variables de salida se han utilizado un conjunto de estadísticas para predecir diferentes indicadores relacionados con las comorbilidades. Estas estadísticas incluyen, la incidencia (INC) y el tiempo hasta evento promedio (TIMETO), además de los intervalos inferiores (L95CI) y superiores (U95CI) de credibilidad al 95%, es decir, los cuantiles 0,025 y 0,975 de los resultados para estos indicadores; también el número de episodios de hipoglucemia graves sufridos durante toda su vida (SHE), el coste (C), los años de vida restantes (LY) y ajustados por calidad (QALY), estos tres últimos, descontados al 3% [11]. Además, se utiliza el porcentaje promedio de afectación de una comorbilidad (UPTO). Estos porcentajes se han separado en intervalos de diez años, desde 20 hasta cien años, y se han evaluado para diferentes edades de la vida (20, 30, 40, 50, 60, 70, 80, 90 y 100 años).

Los datos se han dividido utilizando un rango de 80/20 para crear los conjuntos de entrenamiento y test. El conjunto de entrenamiento se utiliza para entrenar el modelo,

mientras que el conjunto de test se utiliza para evaluar su rendimiento. Esto se hace para asegurar que el modelo no esté sobre ajustado a los datos de entrenamiento y que se generalice bien a datos no vistos.

2.2. Hardware y software

En esta sección se describen los requisitos de hardware y software necesarios para llevar a cabo el proyecto. Esto incluye la descripción del equipo y las librerías utilizadas, así como cualquier otro software o herramienta especial necesaria para el análisis de datos y el entrenamiento y evaluación de modelos.

2.2.1. Hardware

En la realización de este proyecto se ha utilizado el hardware propio del alumno para entrenar y evaluar los modelos de aprendizaje automático subrogados. En concreto, se ha dispuesto de un MacBook Pro de 13 pulgadas, procesador 2.3 GHz Intel Core i5 de doble núcleo, memoria 8 GB 2133 MHz LPDDR3, GPU Intel Iris Plus Graphics 640 1536 MB.

2.2.2. Software

Se ha utilizado un software especializado en el área del aprendizaje automático. Este tipo de software permite implementar y entrenar modelos de aprendizaje automático de manera rápida y sencilla, proporcionando herramientas y librerías que facilitan la tarea. En concreto, se ha utilizado el siguiente:

- Herramientas de programación:
 - El lenguaje de programación Python [20]. Además, se han utilizado las siguientes librerías:
 - Matplotlib: es una librería de Python que permite crear gráficos y visualizaciones de datos de manera sencilla y flexible [12].
 - Numpy: es una librería de Python que proporciona soporte para el cálculo numérico y la manipulación de arrays multidimensionales. Es ampliamente utilizada en el área de la

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

ciencia de datos y el aprendizaje automático debido a sus eficientes funciones y métodos para realizar operaciones matemáticas y estadísticas [17].

- Pandas: es una biblioteca de Python que proporciona herramientas para el análisis y la manipulación de datos tabulares [19].
 - Scikit-learn: es una librería de código abierto en Python que proporciona una amplia variedad de algoritmos de aprendizaje automático y herramientas para su implementación y evaluación. Es una de las librerías más utilizadas en el área del aprendizaje automático y es ideal para principiantes y usuarios avanzados por su sencillez y eficiencia [21].
 - SciPy: es una biblioteca de Python que se utiliza para realizar cálculos científicos y matemáticos avanzados, así como para analizar y procesar datos científicos. Scipy se construye sobre la base de la biblioteca NumPy de Python, y proporciona un conjunto de herramientas para la integración, optimización, álgebra lineal, interpolación, procesamiento de señales, estadísticas y muchas otras tareas de cálculo científico [22].
 - Chart.js: es una librería JavaScript que permite la visualización de datos en gráficos y diagramas interactivos, con una gran variedad de opciones de personalización y adaptabilidad a diferentes tipos de proyectos y datos [3].
- El editor de código fuente Visual Studio Code [28].
- Frameworks utilizados en el prototipo web:
 - Next.js: es un *framework* de React que se utiliza para crear aplicaciones web del lado del servidor y del lado del cliente, lo que permite una mejor experiencia de usuario y un mejor rendimiento en general [16].

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Flask: es un *framework* de Python para construir aplicaciones web que permite crear aplicaciones web de manera rápida y sencilla, con una sintaxis sencilla y una gran flexibilidad [7].
- Herramientas de control de versiones:
 - GitHub: es una plataforma de desarrollo de software que permite almacenar y controlar versiones de código, colaborar en proyectos y compartir código con otros desarrolladores [1].
 - Google Drive [10].
- Redacción de la memoria:
 - Google Docs [9].

2.3. Procedimiento de entrenamiento y evaluación de modelos

En esta sección se describe el procedimiento utilizado para entrenar y evaluar modelos subrogados, además de los algoritmos y las métricas utilizadas para ello.

Se han considerado varias dimensiones importantes en el entrenamiento y evaluación de los modelos subrogados, como dos tipos de experimentos con más y menos parámetros de entrada, tres tipos de modelos simples, agrupados y completos, y cuatro modelos de aprendizaje automático diferentes, incluyendo regresión lineal, árboles de decisión, redes neuronales y modelos de agrupamiento.

Cada dimensión se ha utilizado para evaluar el rendimiento de los modelos en diferentes escenarios y niveles de complejidad en el problema. Todo esto se puede visualizar en la Figura 2.3 que muestra cómo se relacionan estas dimensiones entre sí.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

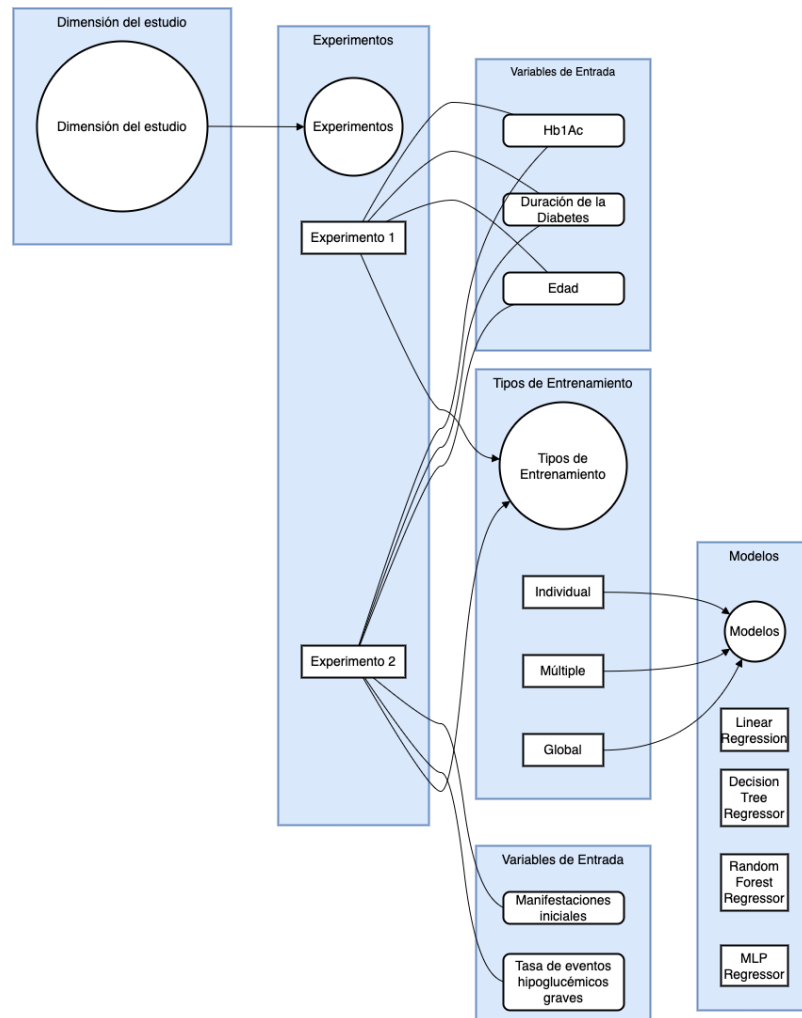


Figura 2.3. Diagrama de la dimensión del estudio.

2.3.1. Entrenamiento de los modelos

En este proyecto se ha implementado la clase *AutoML* para automatizar el proceso de entrenamiento y evaluación de modelos de aprendizaje automático. Esta clase ha sido diseñada para ser una herramienta útil y conveniente para probar diferentes modelos y parámetros de manera rápida y sencilla, encontrar el que mejor se adapte a los datos y tenga el mejor rendimiento [1].

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
 UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

El constructor de la clase divide los datos de entrada y salida en conjuntos de entrenamiento y test utilizando la función *train_test_split* de scikit-learn para los modelos entrenados de forma individual y para el resto de los tipos de entrenamiento se utiliza una recopilación de estos, utilizando la función *get splitted_data*. Luego, se guardan los conjuntos de entrenamiento y test para futuros usos. La explicación detallada de estos tipos se encuentra en la sección 2.3.1.2 de esta memoria.

La estandarización, usando la función *StandardScaler* de scikit-learn. Esta función se basa en dos estadísticas descriptivas de cada característica: la media y la desviación estándar. Primero, se calcula la media de cada característica en el conjunto de datos. Luego, se resta la media de cada valor en la característica correspondiente, lo que centra los datos en cero. A continuación, se calcula la desviación estándar de cada característica y se divide cada valor en la característica por la desviación estándar, lo que escala los datos, lo que ayuda a mejorar la eficiencia y precisión de muchos algoritmos de aprendizaje automático.

Este método toma varios argumentos al inicializarse, incluyendo el modelo, su nombre, el nombre de la clase de la que hereda, el tipo de entrenamiento, los parámetros del modelo y las columnas de entrada y salida. Estas columnas deben contener los datos que se utilizarán para entrenar y evaluar el modelo, y se asume que los datos ya han sido recopilados y preprocesados.

2.3.1.1. Experimentos realizados

En este proyecto se han realizado dos experimentos para evaluar la capacidad predictiva de los modelos subrogados. En el primer experimento se utilizó un conjunto de características de entrada específicas, explicadas en el apartado 2.1 del documento, excluyendo las manifestaciones iniciales.

En el segundo experimento se añadieron estas manifestaciones como características adicionales de entrada, y se comparó el rendimiento de los modelos con y sin estas características.

En ambos casos las variables predichas fueron las mismas que se describieron en el apartado 2.1, es decir, el tiempo promedio hasta evento, el porcentaje promedio de

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

afectación de cada comorbilidad, la incidencia de cada una, los años de vida restantes y ajustados por calidad, el número de episodios hipoglucémicos graves y el coste del tratamiento.

El objetivo de ambos experimentos ha sido evaluar la capacidad de los modelos subrogados para predecir y determinar si la inclusión de características adicionales puede mejorar la precisión del modelo.

2.3.1.2. Tipos de entrenamiento

En este proyecto, se han realizado diferentes tipos de entrenamiento en base a la disposición del conjunto de comorbilidades como se puede ver en la Figura 2.2:

- El tipo de entrenamiento denominado individual o *single* entrena un modelo para predecir el tiempo hasta desarrollar cada comorbilidad mencionada anteriormente.
- El tipo de entrenamiento múltiple entrena un modelo por grupos de comorbilidades para evaluar la relación entre las diferentes enfermedades y su impacto en los indicadores de salida.
- El tipo de entrenamiento global entrena un modelo con todos los grupos de comorbilidades para obtener una predicción general del impacto de éstas en el conjunto estudiado.

Para todos los tipos de entrenamiento la clase *AutoML* utiliza el método *train*. Este método crea un pipeline con el modelo de regresión y sus parámetros y realiza una búsqueda en cuadrícula utilizando la función *GridSearchCV* de scikit-learn para encontrar la mejor combinación de parámetros. Esta función define una cuadrícula de posibles valores para cada hiperparámetro que se desea ajustar. Luego, se entrenan y evalúan modelos utilizando todas las combinaciones de valores de los hiperparámetros en la cuadrícula. Para cada combinación, se realiza una validación cruzada para obtener una estimación precisa del rendimiento del modelo.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

2.3.2. Evaluación de los modelos

Para realizar la evaluación de los modelos es necesario predecir los valores de salida o respuesta que tendrían las entradas de test en los modelos de aprendizaje automático. El método *predict* de la clase realiza esta tarea al utilizar los modelos previamente entrenados para realizar predicciones sobre el conjunto de datos de test.

La evaluación de estos modelos se realiza comparando los valores predichos con los valores verdaderos en el conjunto de testeo utilizando una métrica de rendimiento, como el coeficiente de determinación (R^2) o la pérdida de regresión del error porcentual absoluto medio (MAPE) [4] [13]. Cabe destacar que la evaluación del rendimiento de cada indicador para cada comorbilidad en cada modelo del Experimento 2 se calcula despreciando las réplicas que tienen esa comorbilidad como variable de entrada en el conjunto de test y predicción. Esta distribución se puede ver en la Tabla 2.1.

Comorbilidad	Nº de registros (antes)	Nº de registros (después)
Fallo Cardíaco	4212	3704
Infarto de Miocardio	4212	3705
Angina	4212	3688
Ictus	4212	3677
Ceguera	4212	3700
Edema macular diabético	4212	3540
Retinopatía de fondo	4212	3620
Retinopatía proliferativa	4212	3636
Neuropatía	4212	3537
Amputación extremidades inferiores	4212	3523
Microalbuminuria	4212	3619
Macroalbuminuria	4212	3603
Enfermedad renal terminal	4212	3635

Tabla 2.1. Relación del número de registros del conjunto de test y predicción antes y después de despreciar los registros con esa comorbilidad como variable de entrada.

De esta forma, se calcula el desempeño de cada modelo mediante el uso de dos métricas específicas, las cuales se detallan con mayor profundidad en la sección 2.3.4. Además, se llevó a cabo una comparación entre los modelos entrenados de manera individual,

múltiple y global. Esta evaluación permitió determinar si existe alguna ventaja en la capacitación de modelos específicos para cada enfermedad, o si el modelo global es lo suficientemente general como para predecir con precisión la evolución de cualquier complicación.

En cualquier caso, si se observa que el rendimiento del modelo no es satisfactorio, es posible realizar ajustes en los hiper-parámetros y volver a entrenar el modelo para mejorar su rendimiento. Este proceso puede repetirse hasta que el rendimiento del modelo sea aceptable.

2.3.3. Algoritmos utilizados

En este proyecto se han utilizado los siguientes algoritmos de regresión de aprendizaje automático, pertenecientes a la librería de Scikit-Learn:

- *Linear Regression*: que es un tipo de modelo de aprendizaje automático utilizado para predecir una variable continua a partir de una o más variables explicativas [24].
- *Random Forest Regressor*: que combina múltiples árboles de decisión para crear un modelo más preciso para la regresión, con el objetivo de evaluar cuál de ellos ofrece el mejor rendimiento para cada tipo de modelo y seleccionar el que obtenga mejores resultados [23].
- *Decision Tree Regressor*: divide el conjunto de datos en conjuntos cada vez más pequeños en función de los valores de las variables definidas hasta que se alcanza una condición de parada predefinida, y luego predice la variable objetivo utilizando el valor promedio de la variable objetivo en la sección correspondiente [26].
- *Multi-layer Perceptron Regressor*: utiliza una red neuronal artificial con múltiples capas ocultas para predecir la variable objetivo. Durante el entrenamiento, ajusta los pesos y sesgos de las neuronas para reducir la función de pérdida y mejorar la precisión de las predicciones [25].

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

2.3.4. Métricas utilizadas

Para evaluar el rendimiento de los modelos, se han utilizado dos métricas comunes: el coeficiente de determinación (R^2) y la pérdida de regresión del error porcentual absoluto medio (MAPE).

El coeficiente de determinación (R^2) es una medida comúnmente empleada para evaluar el desempeño de los modelos de predicción. Este coeficiente indica la proporción de la variabilidad de la variable dependiente que puede ser explicada por el modelo. Su fórmula se describe a continuación:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad | \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \quad \wedge \quad \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n \epsilon_i^2$$

Un valor de R^2 cercano a 1 indica que el modelo es capaz de explicar de manera efectiva la variación de esta variable, lo que indica una buena capacidad predictiva. Por otro lado, un valor de R^2 cercano a 0 indica que el modelo no es capaz de explicar adecuadamente la variación de la variable objetivo, lo que sugiere una baja capacidad predictiva.

El error porcentual absoluto medio, de sus siglas en inglés MAPE, es una métrica que mide el error porcentual promedio entre las predicciones de un modelo y los valores reales observados en los datos de test. Se calcula como la media de los errores porcentuales absolutos para cada predicción, como figura a continuación:

$$MAPE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} \frac{|y_i - \hat{y}_i|}{\max(\epsilon, |y_i|)}$$

Al expresar el error en términos porcentuales, el MAPE permite comparar modelos con diferentes escalas de valores y es una herramienta útil para evaluar la calidad de las predicciones de un modelo.

Estas métricas han permitido comparar el rendimiento de los modelos con los obtenidos por el modelo de simulación y determinar cuál es la opción más precisa y adecuada para nuestra aplicación. Cabe destacar que cada métrica se calcula a través de la salida

promedio y sus dos intervalos de confianza que brindan información sobre la precisión, la incertidumbre y la variabilidad de las estimaciones.

2.4. Conexión de los modelos con la aplicación

En este proyecto, se ha dado importancia a la integración entre la aplicación web y los modelos de aprendizaje automático subrogados, y se ha realizado un estudio de la estructura de los objetos JSON para implementar correctamente la integración.

La aplicación web envía un objeto JSON, cuya estructura se puede ver en la Figura 2.4, al modelo de simulación cuando se ejecuta una solicitud de datos. Este objeto es recibido por el modelo, que lo transforma y ejecuta dos simulaciones: la primera para simular un paciente base y la segunda para simular el desarrollo con el posible tratamiento. Después de esto, se genera un nuevo objeto JSON que contiene los resultados de ambas simulaciones y se envía de vuelta a la aplicación web.

```
{
  "patient": {
    "man": "false",
    "age": "40",
    "durationOfDiabetes": "5",
    "hypoRate": "2",
    "hypoRateRR": "0.5",
    "baseHbA1cLevel": "8",
    "objHbA1cLevel": "8",
    "annualCost": "3000",
    "manifestations": [
      "ANGINA",
      "NEU"
    ]
  }
}
```

Figura 2.4. Ejemplo de objeto JSON enviado por la aplicación

Este objeto JSON tiene una variable llamada *patient* que a su vez contiene varias variables que describen las características del paciente. Las variables que se encuentran en el objeto son:

- *man*: indica si el paciente es hombre (*true*) o mujer (*false*).
- *age*: edad del paciente en años.

- *durationOfDiabetes*: tiempo desde el diagnóstico de diabetes en años.
- *hypoRate*: número de episodios de hipoglucemia al año.
- *hypoRateRR*: tasa de riesgo relativo para episodios de hipoglucemia.
- *baseHbA1cLevel*: nivel de HbA1c.
- *objHbA1cLevel*: nivel objetivo de HbA1c para el paciente.
- *annualCost*: costo anual estimado del tratamiento.
- *manifestations*: una lista que contiene las manifestaciones del paciente.

Para que los modelos de aprendizaje automático subrogados funcionen de la misma forma, se ha desarrollado un script que permite que estos tipos de modelos reciban los objetos JSON de la misma manera que el modelo de simulación. Esto es necesario para que la integración entre la aplicación web y los modelos de aprendizaje automático subrogados sea lo más fluida posible y que no haya problemas de compatibilidad o comunicación entre ambas partes.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Capítulo 3 -

3. Resultados

En este capítulo se detallan los resultados de la elección de la métrica de evaluación, los modelos entrenados utilizando los diferentes tipos de entrenamiento y midiendo su rendimiento para cada modelo, comorbilidad y agrupación mediante tablas y una serie de gráficas que ayudan a la visualización de los resultados obtenidos.

3.1. Elección de las métricas de evaluación

En este estudio se evaluaron varios modelos de predicción de la variable objetivo, utilizando dos métricas de evaluación, R^2 y MAPE. Como veremos en los siguientes epígrafes, los resultados mostraron que los modelos de predicción desarrollados en este estudio tuvieron valores de R^2 satisfactorios, lo que indica que pueden explicar efectivamente la variabilidad de la variable objetivo.

Sin embargo, se encontró que el MAPE no era adecuado para evaluar el desempeño de los modelos en situaciones donde los valores reales de la variable objetivo son cercanos a cero, esto ocurre en el caso de algunos indicadores como la incidencia o el porcentaje promedio de afectación.

Modelo	Individual	Múltiple	Global
LinearRegression	2,97198E+15	2,97198E+15	2,97198E+15
RandomForestRegressor	16965079690293	5884185865702	105979022492250
DecisionTreeRegressor	4812777585615	5811478685487	21583745515484
MLPRegressor	151257572606756	614780850147378	1,9575E+15

Tabla 3.1. Resultados de MAPE para el porcentaje promedio de afectación de la ceguera.

Esto se debe a que el MAPE puede producir resultados muy grandes o incluso infinitos en esas situaciones, cómo se refleja en la Tabla 3.1, lo que afecta la interpretación y

comparación de los resultados y podría afectar la toma de decisiones basadas en el análisis.

Por lo tanto, se decidió descartar el MAPE como métrica de evaluación y se optó por utilizar únicamente el coeficiente de determinación R^2 . En el capítulo 4, se discutirá en detalle la elección de ésta métrica y cómo contribuyó a fundamentar las conclusiones y decisiones basadas en los resultados del análisis.

3.2. Tipos de entrenamiento

En este epígrafe se describen los resultados obtenidos por los distintos tipos de entrenamiento realizados en los modelos de predicción.

3.2.1. Experimento sin manifestaciones iniciales

Los resultados obtenidos para determinar el tiempo de promedio hasta la aparición e incidencia de las manifestaciones sin utilizar estas como una variable de entrada para los modelos mostraron que el tipo de entrenamiento global es el que mejores resultados produce en términos de R^2 para estos indicadores. En las Tablas 3.2 y 3.3 se muestran los resultados para dos de las comorbilidades estudiadas.

Modelo	Individual	Múltiple	Global
LinearRegression	0,9043840994	0,9043840994	0,9043840994
RandomForestRegressor	0,9935514042	0,9934281602	0,9920871004
DecisionTreeRegressor	0,9934653511	0,9939682549	0,9939864265
MLPRegressor	0,9949739931	0,9954749374	0,9959084757

Tabla 3.2. Experimento 1: Resultados de R^2 del tiempo promedio hasta aparición del infarto de miocardio, según forma de entrenamiento. En verde, el mejor valor.

En cambio, los modelos entrenados para predecir el porcentaje promedio de afectación de una manifestación se ajustaron mejor para los grupos de comorbilidades, es decir, para el entrenamiento de forma múltiple, como se ve en la Tabla 3.4.

Modelo	Individual	Múltiple	Global
LinearRegression	0,9600908096	0,9600908096	0,9600908096
RandomForestRegressor	0,9949802357	0,9948853746	0,9925019893
DecisionTreeRegressor	0,9954463442	0,9953334392	0,9710232631
MLPRegressor	0,996797588	0,9968181579	0,9969142509

Tabla 3.3. Experimento 1: Resultados de R^2 de la incidencia de la amputación de extremidades inferiores, según forma de entrenamiento. En verde, el mejor valor.

Modelo	Individual	Múltiple	Global
LinearRegression	0,4617684924	0,4617684924	0,4617684924
RandomForestRegressor	0,9736619801	0,9735190222	0,9723496566
DecisionTreeRegressor	0,9793375663	0,9795312667	0,9788219636
MLPRegressor	0,8467382909	0,8188620004	0,8210698751

Tabla 3.4. Experimento 1: Resultados de R^2 del porcentaje promedio de afectación del edema macular diabético, según forma de entrenamiento. En verde, el mejor valor.

3.2.2. Experimento con manifestaciones iniciales

Los hallazgos encontrados al predecir el tiempo promedio hasta que se manifiesta alguna comorbilidad y su incidencia, indicaron que el entrenamiento *single* es la técnica que arroja los mejores resultados en cuanto al valor de R^2 para los modelos que pronostican dicho indicador, a diferencia de lo que resultó del apartado anterior. Las Tablas 3.5 y 3.6 presentan los resultados de estos indicadores.

Por el contrario, al igual que ocurrió en el experimento sin manifestaciones iniciales, los modelos diseñados para prever el porcentaje medio de individuos afectados por una manifestación tuvieron una mejor precisión al ser entrenados de forma múltiple. Esta observación se puede apreciar en la Tabla 3.7.

Modelo	Individual	Múltiple	Global
LinearRegression	0,7292314838	0,7292314838	0,7292314838
RandomForestRegressor	0,9964601137	0,9964937946	0,9785968291
DecisionTreeRegressor	0,9960721657	0,9962328787	0,9580386069
MLPRegressor	0,997634131	0,9974299834	0,9969779541

Tabla 3.5. Experimento 2: Resultados de R^2 del tiempo promedio hasta aparición de retinopatía proliferativa, según forma de entrenamiento. En verde, el mejor valor.

Modelo	Individual	Múltiple	Global
LinearRegression	0,8944602797	0,8944602797	0,8944602797
RandomForestRegressor	0,9991192367	0,9987285628	0,9806305038
DecisionTreeRegressor	0,9987837031	0,9981038032	0,9947493393
MLPRegressor	0,999402711	0,9993431278	0,9990875096

Tabla 3.6. Experimento 2: Resultados de R^2 de la incidencia de microalbuminuria, según forma de entrenamiento. En verde, el mejor valor.

Modelo	Individual	Múltiple	Global
LinearRegression	0,3357361203	0,3357361203	0,3357361203
RandomForestRegressor	0,9880307179	0,9885354706	0,801176262
DecisionTreeRegressor	0,9862129953	0,9868506765	0,7869406469
MLPRegressor	0,8604588041	0,8466266397	0,5964040945

Tabla 3.7. Experimento 2: Resultados de R^2 del porcentaje promedio de afectación de neuropatía, según forma de entrenamiento. En verde, el mejor valor.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

3.3. Ajuste de los modelos

Analizando los diferentes modelos entrenados se ha observado, en los diferentes experimentos realizados, que los intervalos de confianza generados para las predicciones se ajustaron peor que la salida promedio. Esto puede verse en la Figura 3.1, donde se muestra una dispersión de los valores reales y predichos del tiempo promedio hasta la aparición de una de las comorbilidades estudiadas, en este caso, la ceguera mediante el entrenamiento de tipo *single* y el algoritmo de *Linear Regression*.

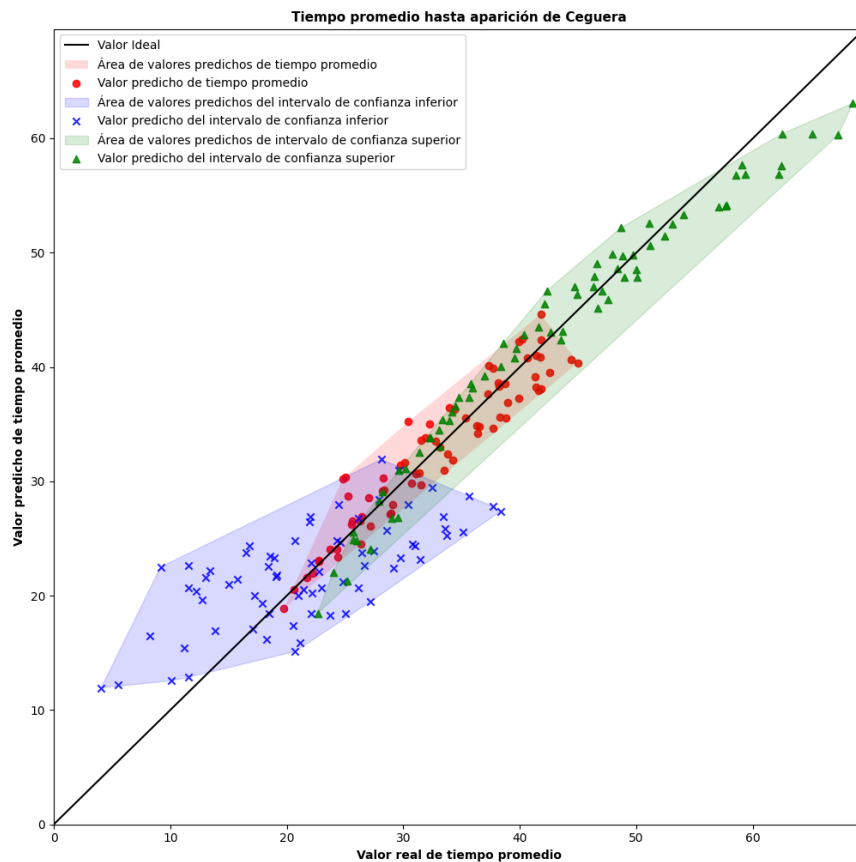


Figura 3.1. Experimento 1: Dispersión del tiempo promedio hasta aparición de la ceguera, entrenamiento de tipo *single* mediante *Linear Regression*.

El mal ajuste de los intervalos de confianza puede tener varias causas, entre las que se incluyen la complejidad del modelo, la calidad de los datos y la cantidad de datos

disponibles en los diferentes experimentos realizados. Estos factores se profundizan más en el capítulo 4.

En general, estos problemas pueden tener un impacto negativo en la precisión de las estimaciones y, por lo tanto, en los intervalos de confianza. Es importante tener en cuenta estos factores al realizar inferencias a partir de los datos y seleccionar los modelos más adecuados para el análisis.

Además, es importante destacar que el ajuste de los intervalos de confianza es crucial en el análisis de datos, ya que proporcionan una medida de incertidumbre en las predicciones y estimaciones.

3.4. Comparación de los algoritmos utilizados

Los resultados de la comparación de los distintos algoritmos explicados en el apartado 2.3.3 indicaron que *MLP Regressor* fue el algoritmo que arrojó los mejores resultados para las métricas de tiempo medio del evento y morbilidad, en ambos ensayos.

En otras palabras, *MLPRegressor* pudo predecir con mayor precisión el tiempo promedio hasta que ocurrió el evento y la incidencia de dicho evento, en comparación con el resto de algoritmos utilizados. Para ilustrar esto se muestra la Figura 3.2 como ejemplo de la dispersión de valores reales frente a predichos del tiempo promedio hasta aparición de una de las comorbilidades estudiadas:

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015. Su autenticidad puede ser contrastada en la siguiente dirección https://sede.ull.es/validacion/	
Identificador del documento: 5429785	Código de verificación: nPGy9YjY
Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

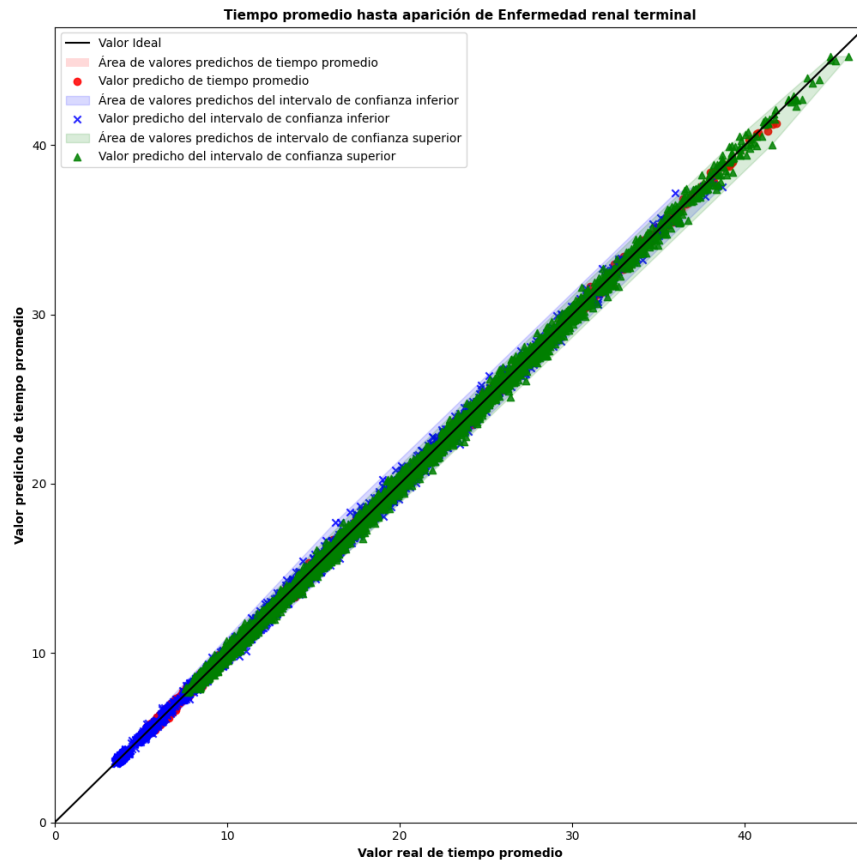


Figura 3.2. Experimento 2: Dispersión del tiempo promedio hasta aparición de la enfermedad renal terminal, entrenamiento de tipo *single* mediante *MLP Regressor*.

Sin embargo, los resultados también indicaron que el mejor algoritmo para predecir el porcentaje de afectación es el *Decision Tree Regressor* en el Experimento 1, mientras que para el Experimento 2 es el *Random Forest Regressor*. Esto indica que estos algoritmos pueden predecir con mayor precisión este indicador, como se muestra en la Figura 3.3, que muestra una dispersión de los valores reales y predichos de este indicador para ambos experimentos:

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
 UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

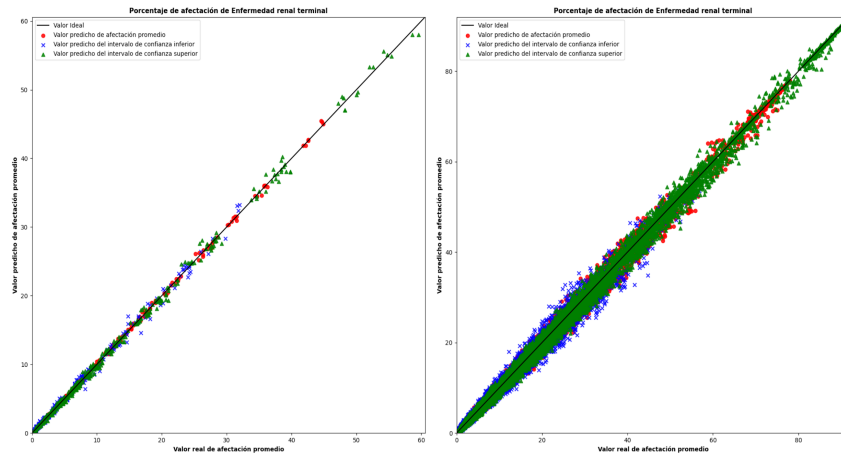


Figura 3.3. Experimento 1 a la izquierda y Experimento 2 a la derecha: Dispersión del tiempo promedio hasta aparición de la enfermedad renal terminal, entrenamiento de tipo *single* mediante *MLPRegressor*.

3.5. Análisis de los modelos subrogados

En este apartado, se presenta el análisis de los modelos subrogados desarrollados en el estudio. Se discutirá la sobreestimación observada en el grupo de enfermedades cardíacas y se analizarán las diferencias de tiempo entre el modelo de simulación y los modelos subrogados y la robustez y fiabilidad de estos.

3.5.1. Sobreestimación y difícil ajuste en el grupo de enfermedades cardíacas

Los modelos subrogados desarrollados en este estudio mostraron una precisión y un desempeño prometedores en la predicción del porcentaje de afectación de una comorbilidad. Sin embargo, se observó una tendencia en los modelos a sobreestimar las predicciones en el grupo de enfermedades cardíacas, esto se puede ver en la Figura 3.4 que muestra la relación de afectación de las manifestaciones pertenecientes a esta agrupación.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

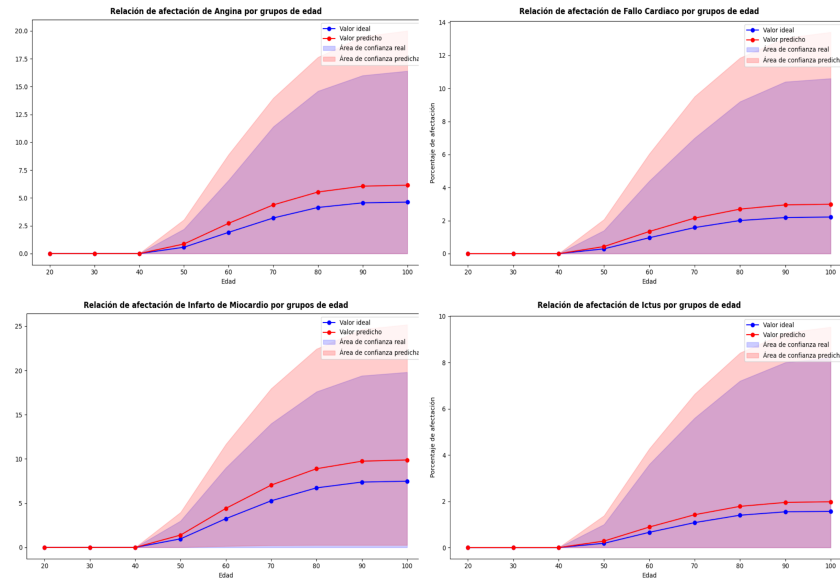


Figura 3.4. Experimento 2: Gráfica del porcentaje de afectación de enfermedades cardíacas, entrenamiento de tipo múltiple mediante *Random Forest Regressor*.

Esta sobreestimación está ligada a un pobre ajuste de este grupo de comorbilidades como puede verse en la Figura 3.5 que muestra la dispersión de una de estas comorbilidades para el Experimento 2, utilizando el algoritmo que mejor se ajusta explicado en el apartado 3.4.

A pesar de esta tendencia, no se encontró un patrón claro de sobreestimación ni de infraestimación en los demás grupos de comorbilidades estudiados. En el capítulo 4 del estudio se discute la posible causa de esta sobreestimación en el grupo de enfermedades cardíacas.

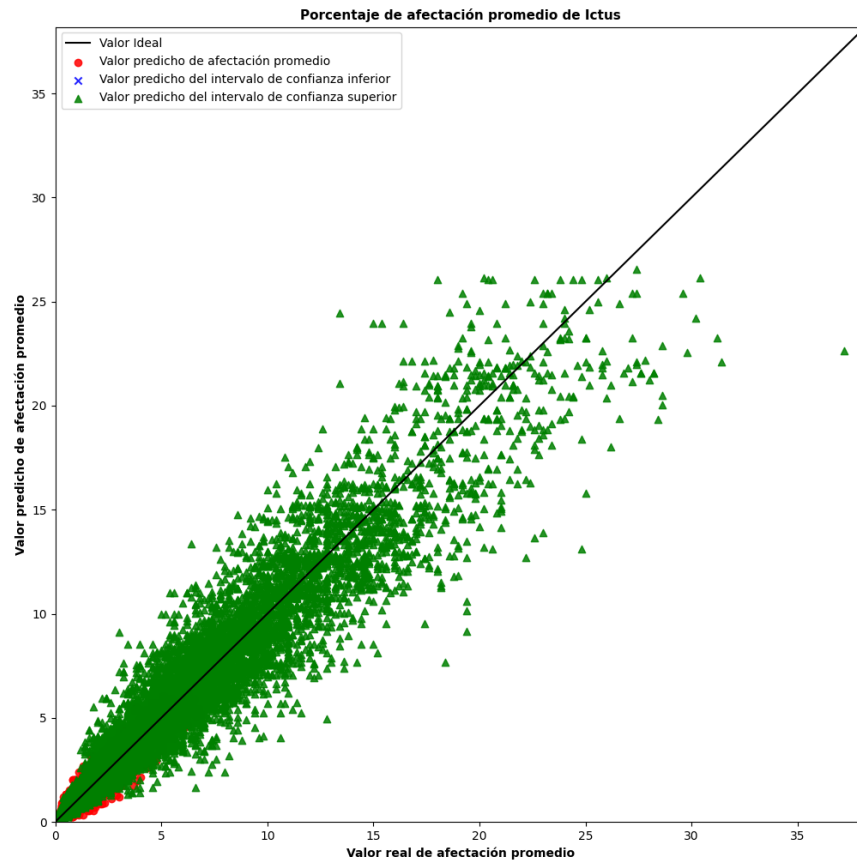


Figura 3.5. Experimento 2: Dispersión del porcentaje de afectación del ictus, entrenamiento de tipo múltiple mediante *Decision Tree Regressor*.

3.5.2 Implicaciones de las manifestaciones iniciales

En esta sección, se presentan los resultados que involucran la aparición o no de manifestaciones como características de entrada de los modelos subrogados.

En particular, los modelos que utilizan manifestaciones iniciales como datos de entrada muestran una reducción significativa en el tiempo de aparición de comorbilidades relacionadas, es decir, las que pertenecen al mismo grupo de comorbilidades. La Figura 3.6 muestran el porcentaje de afectación en una réplica de un varón de 50 años con 30

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

años de duración de la diabetes, una tasa anual de episodios graves de hipoglucemia con valor 2 y un nivel de Hb1Ac de 11, sin y con retinopatía de fondo respectivamente.

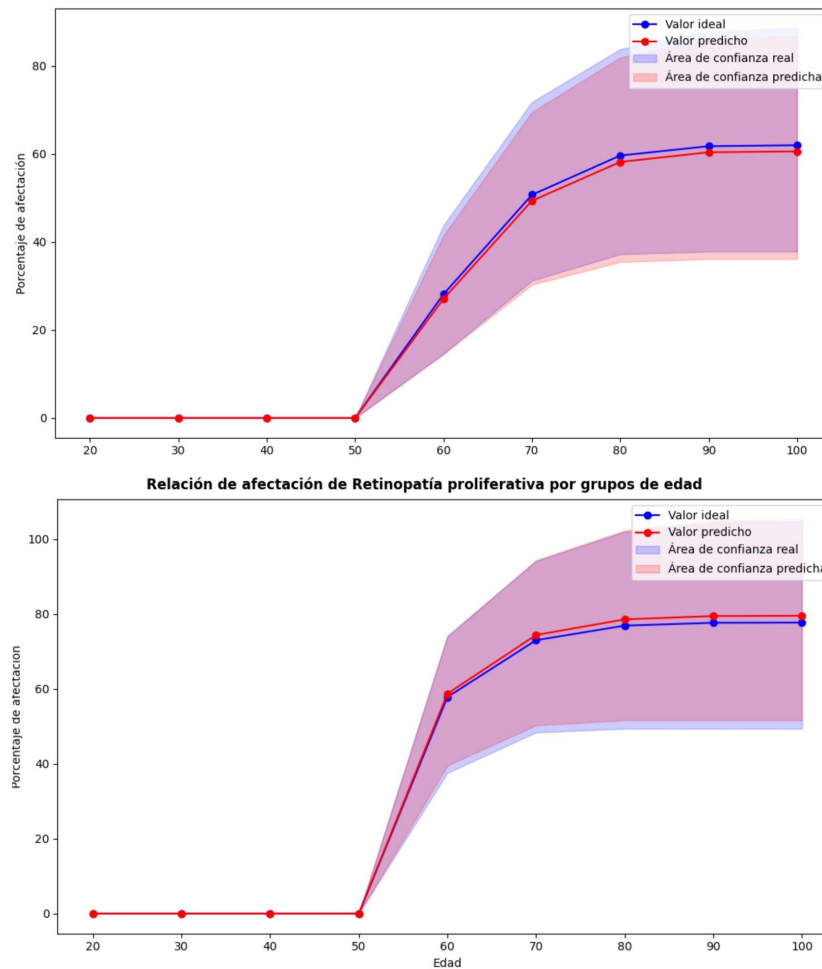


Figura 3.6. Experimento 2: Gráfica del porcentaje de afectación de retinopatía proliferativa, sin y con retinopatía de fondo, entrenamiento de tipo múltiple mediante *Random Forest Regressor*.

Esta gráfica muestra que en la réplica sin ninguna manifestación inicial, la relación de afectación de la retinopatía proliferativa aumentó paulatinamente en comparación con la réplica que sí sufrió retinopatía de fondo. Además, en la gráfica que representa la réplica sin manifestaciones iniciales, el porcentaje afectado creció de forma mucho más lenta,

alcanzando un 60% a los 80 años. En contraste, en la gráfica que representa la réplica con retinopatía de fondo, el porcentaje aumentó de forma exponencial, alcanzando un 80% a esa misma edad.

3.5.3. Aceleración en los tiempos de cómputo

El uso de modelos subrogados tiene una ventaja significativa en términos de tiempo de ejecución en comparación con el modelo de simulación. Esta reducción en el tiempo de ejecución es posible debido a que los modelos subrogados utilizan técnicas estadísticas avanzadas y técnicas de aprendizaje automático para simplificar y generalizar el modelo de simulación original. Como resultado, los modelos subrogados son capaces de generar respuestas rápidas y precisas, lo que los convierte en una herramienta valiosa para la toma de decisiones en tiempo real.

Para ilustrar la ventaja se compararon los tiempos de ejecución de ambos enfoques, en el caso del modelo de simulación tuvo una media de 5.4 segundos para realizar la simulación utilizada como conjunto de datos, mientras que los modelos subrogados tienen un promedio de 0,002340568066 segundos para las predicciones equivalentes. Como se puede observar, los modelos subrogados tienen tiempos de ejecución significativamente más bajos que los modelos de simulación en todos los casos, lo que demuestra que el objetivo principal de este proyecto se ha cumplido.

Esta ventaja en el tiempo de ejecución es de gran importancia en la práctica, ya que permite una mayor eficiencia en el proceso de toma de decisiones en tiempo real. Además, la reducción del tiempo de ejecución permite una mayor capacidad de procesamiento, lo que permite la exploración de una mayor cantidad de escenarios y la realización de un análisis más detallado.

3.6. Conexión de los modelos con la aplicación

En esta sección se presenta un prototipo web implementado por el alumno, como paso previo a la integración con la aplicación final explicada en la sección 1.3.4.

El prototipo web implementado permite a los usuarios cargar sus datos de pacientes en un formulario como se puede ver en la Figura 3.7 y permite obtener predicciones de

riesgo de las comorbilidades utilizando los modelos de aprendizaje automático subrogados desarrollados en este estudio.

Inicio

Sexo: Masculino (dropdown) Edad: 40 Duración de la diabetes (en años): 24

Nivel Base de HbA1c (%): 12 Nivel Objetivo de HbA1c (%): 10

Episodios de hipoglucémicos severos (al año): 3 Objetivo de episodios de hipoglucémicos severos (al año): 2

Costo de tratamiento (en dólares): 230300

Manifestaciones clínicas:

<input checked="" type="checkbox"/> Retinopatía de Fondo	<input type="checkbox"/> Retinopatía Proliferativa
<input type="checkbox"/> Edema Macular Diabético	<input type="checkbox"/> Enfermedad Renal Terminal
<input type="checkbox"/> Ceguera	<input type="checkbox"/> Microalbuminuria
<input type="checkbox"/> Macroalbuminuria	<input type="checkbox"/> Ictus
<input type="checkbox"/> Angina	<input type="checkbox"/> Fallo Cardíaco
<input type="checkbox"/> Infarto de Miocardio	<input type="checkbox"/> Amputación de Extremidades Inferiores
<input type="checkbox"/> Neuropatía	

Enviar

Figura 3.7. Formulario del prototipo web.

Este emulador, utilizando el framework de Next.js para el frontend y Flask para el backend, genera una serie de gráficos que muestran el desarrollo de las enfermedades presentadas en este estudio, mediante la librería de Chart.js. La Figura 3.8 muestra estos gráficos luego de rellenar el formulario.

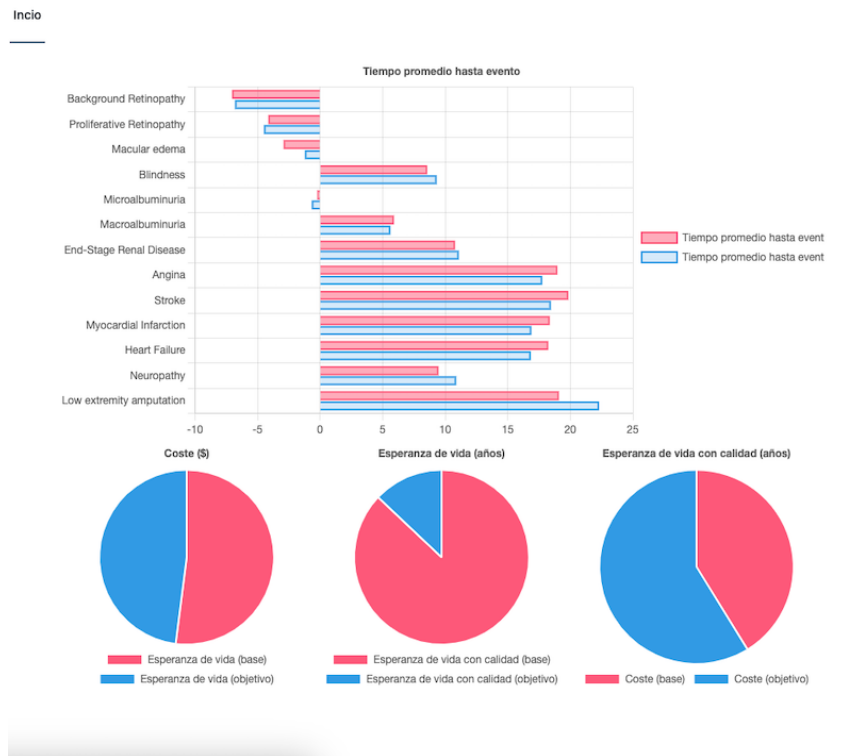


Figura 3.8. Gráficos de los valores del riesgo de las comorbilidades.

Cabe destacar que este prototipo ha sido implementado con el objetivo de ser un emulador por lo que respeta el formato JSON requerido para el intercambio de mensajes, presentado en la sección 2.4, así pues la conexión con la aplicación puede ser realizada en cualquier momento.

Se espera que esta herramienta pueda ayudar a los profesionales de la salud y los pacientes a tomar decisiones informadas sobre el tratamiento y la gestión de la comorbilidades presentadas y también facilitar la investigación en este campo.

- Capítulo 4 -

4. Discusión

En este capítulo 4 se aborda la discusión en torno a la elección del coeficiente de determinación (R^2) como métrica de evaluación en modelos de regresión. Además, se presenta un análisis del ajuste de los modelos y los posibles factores que pueden afectarlos, como la complejidad del modelo, la calidad de los datos y la cantidad de datos disponibles. También se examina la sobreestimación de enfermedades cardíacas en el modelo de simulación y se propone una hipótesis para explicar este fenómeno. Por último, se discuten limitaciones en la evaluación de la efectividad de los tipos de entrenamiento y posibles alternativas.

4.1. Elección del coeficiente de determinación

En esta sección, se discute la elección del coeficiente de determinación (R^2) como métrica de evaluación.

El R^2 es una medida comúnmente utilizada en la evaluación de modelos de regresión que proporciona una medida de cuánta varianza en los datos de respuesta es explicada por el modelo. Un valor de R^2 cercano a 1 indica que el modelo es capaz de explicar una gran proporción de la variabilidad en los datos de respuesta, lo que sugiere un buen ajuste del modelo a los datos. Por lo tanto, el R^2 es una métrica intuitiva y fácil de interpretar, lo que facilita la comunicación de los resultados del modelo a los interesados.

Además, el R^2 es menos sensible a los errores absolutos, como el MAPE, que puede dar lugar a valores altos incluso cuando los errores son pequeños en términos relativos, como se explicó en el apartado 3.1. En cambio, el R^2 es una medida de ajuste relativa que no se ve afectada por los valores absolutos de los errores, lo que lo hace más adecuado para evaluar modelos con datos de diferentes magnitudes.

4.2. Ajuste de los modelos

En el apartado 3.3, se presentan los resultados del ajuste de los modelos a los datos de los experimentos realizados.

El mal ajuste de los intervalos de confianza puede deberse a varios factores, entre los cuales destacan la complejidad del modelo, la calidad de los datos y la cantidad de datos disponibles en los diferentes experimentos realizados. En cuanto a la complejidad del modelo, se ha observado que modelos muy simples, como el de regresión lineal expuesto en la Figura 4.1, pueden ajustarse mal a los datos de entrenamiento y, por lo tanto, pueden tener un mal desempeño en datos nuevos.

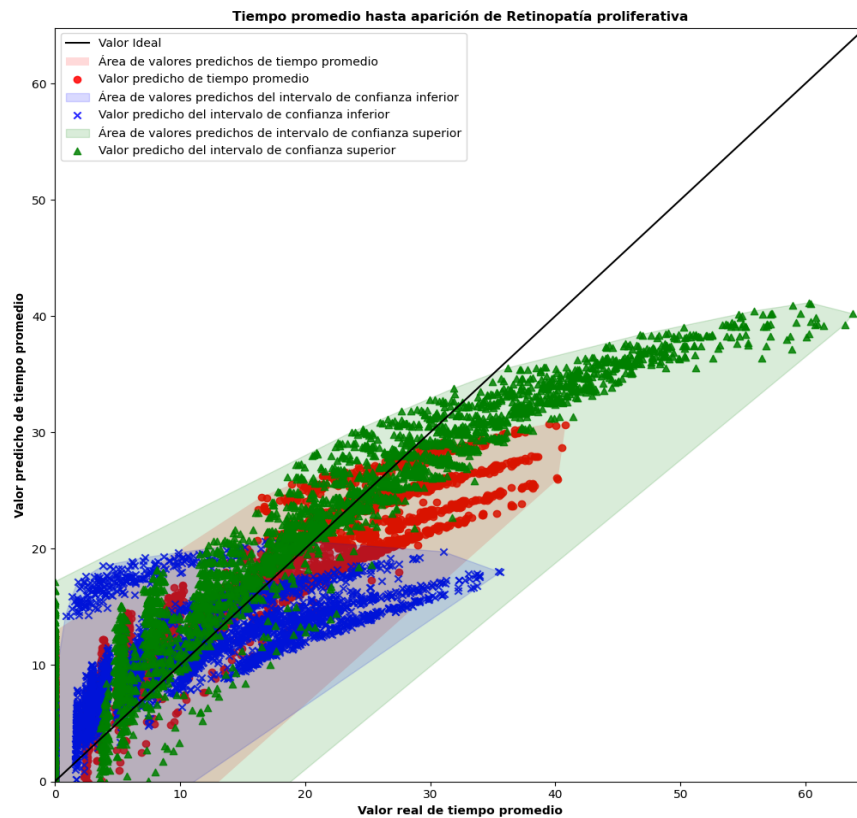


Figura 4.1. Experimento 2: Dispersión del tiempo promedio hasta aparición de la retinopatía proliferativa, entrenamiento de tipo global mediante *Linear Regression*.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

Otro factor que puede afectar el ajuste de los intervalos de confianza es la calidad de los datos. Es importante tener en cuenta que al utilizar los datos generados por un modelo de simulación para realizar predicciones y explorar escenarios hipotéticos, los resultados obtenidos pueden diferir de los resultados reales.

Por último, la cantidad de datos disponibles también puede tener un impacto en el ajuste de los intervalos de confianza. En general, cuanto mayor sea la cantidad y calidad de datos disponibles, mejor será el desempeño del modelo. Sin embargo, ha sido difícil obtener una cantidad suficiente de datos de alta calidad, esto es debido al gran tiempo de inferencia que tiene el modelo de simulación utilizado, lo que puede limitar la capacidad de los intervalos para ajustarse a los datos.

4.3. Sobreestimación de enfermedades cardíacas

En esta sección, se examina la posible causa de la sobreestimación, presentada en la sección 3.5.1, en el grupo de enfermedades cardíacas y se propone una hipótesis que podría estar contribuyendo a este fenómeno.

La enfermedad cardíaca es una enfermedad crónica y progresiva que puede tener múltiples eventos a lo largo del tiempo. Desafortunadamente, el modelo de simulación se diseñó con la simplificación de que «contempla únicamente el primer evento, sea este una angina, un ictus, un fallo cardíaco o un infarto de miocardio», es decir, el modelo sortea entre las enfermedades del grupo de manera equiprobable y sin considerar las características individuales de cada paciente, como se puede ver en la Figura 4.2, lo que lleva a sobreestimar el riesgo de la enfermedad en los pacientes [5].

Esta limitación puede tener implicaciones significativas en la toma de decisiones clínicas y en la evaluación de los costos asociados con el tratamiento de la enfermedad cardiovascular. Además, también afecta a la capacidad de los modelos de aprendizaje automático para ajustarse adecuadamente a los datos de entrenamiento y hacer predicciones precisas. En consecuencia, los resultados de los modelos de aprendizaje automático subrogados utilizados son menos precisos y, por lo tanto, menos útiles para apoyar la toma de decisiones clínicas y la gestión de la enfermedad cardiovascular.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

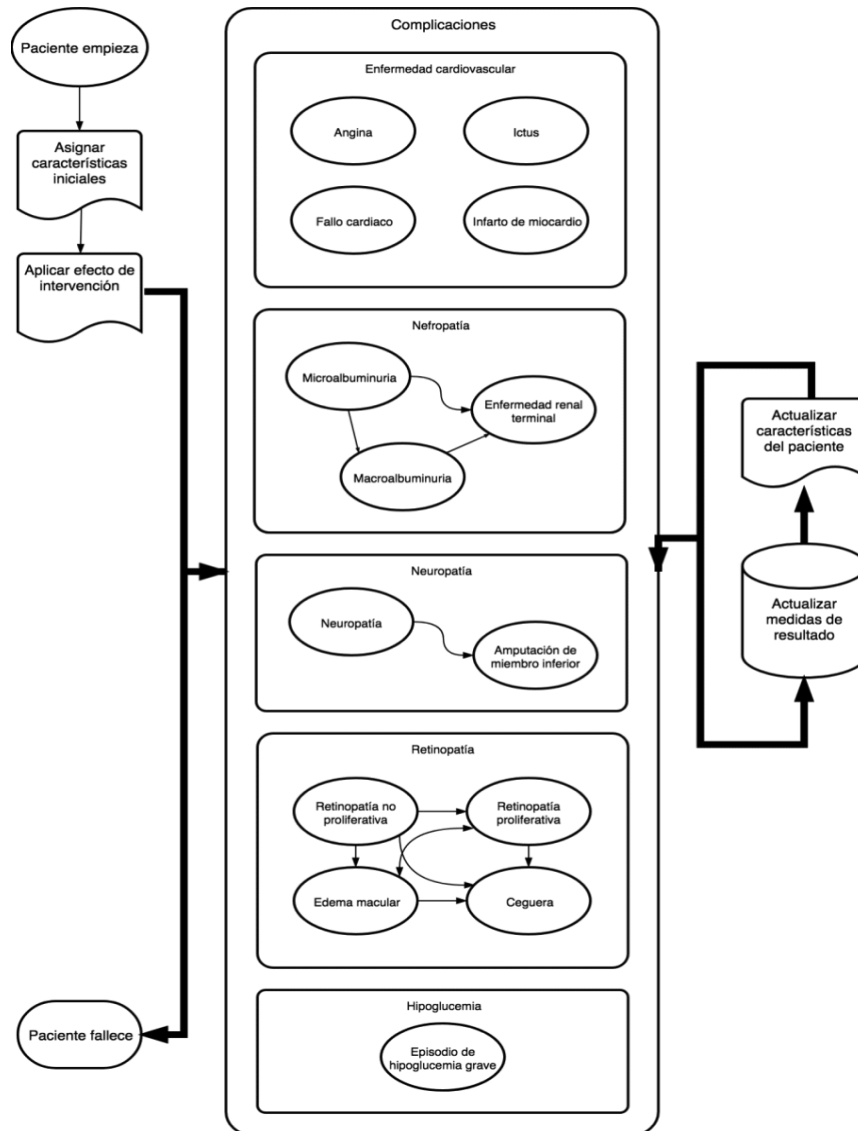


Figura 4.2. Descripción general del modelo de simulación [5].

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
 UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
 UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

- Capítulo 5 -

5. Conclusiones

En este proyecto se ha logrado adaptar modelos de aprendizaje automático a un modelo de simulación utilizando modelos subrogados. Como se ha mostrado en el capítulo 3, esto ha permitido realizar predicciones de forma más rápida sin que exista una pérdida notable de precisión en comparación con el modelo de simulación descrito, lo que es especialmente útil en situaciones donde el modelo de simulación es muy lento o costoso de ejecutar.

Se han seleccionado y entrenado varios modelos de aprendizaje automático utilizando técnicas y algoritmos adecuados para el problema en cuestión. Los resultados de la evaluación muestran que para el primer experimento se ha obtenido un R^2 de 0,9984340817 de máximo y de 0,999294901 para el segundo experimento.

Después de la evaluación y comparación de los resultados obtenidos por los modelos de aprendizaje automático, se ha seleccionado los modelos que proporcionan mejores resultados para ser conectados al prototipo de aplicación web.

En el futuro, se deberían ajustar estos modelos a partir de datos reales para mejorar aún más su precisión y capacidad de predicción en estas situaciones. También se podría explorar la posibilidad de utilizar técnicas de aprendizaje por refuerzo para optimizar los resultados y mejorar la capacidad de toma de decisiones de los modelos.

En definitiva, el uso de modelos de aprendizaje automático subrogados como alternativa a los modelos de simulación ofrece un gran potencial para mejorar la eficiencia y la precisión de los modelos, lo que puede tener implicaciones importantes en una amplia gama de áreas, desde la ingeniería hasta la gestión de recursos y la toma de decisiones.

- Capítulo 6 -

6. Conclusions

In this project we have been able to adapt *machine learning* models to a simulation model using surrogate models. As shown in Chapter 3, this has allowed faster predictions to be made without a noticeable loss of accuracy compared to the simulation model described, which is especially useful in situations where the simulation model is very slow or costly to run.

Several *machine learning* models have been selected and trained using techniques and algorithms suitable for the problem at hand. The evaluation results show that for the first experiment an R^2 of 0.9984340817 maximum and 0.999294901 for the second experiment have been obtained.

After the evaluation and comparison of the results obtained by the *machine learning* models, the models providing the best results have been selected to be connected to the web application prototype.

In the future, these models should be adjusted from real data to further improve their accuracy and predictive ability in these situations. The possibility of using reinforcement learning techniques could also be explored to optimize the results and improve the decision-making capability of the models.

In short, the use of surrogate *machine learning* models as an alternative to simulation models offers great potential for improving the efficiency and accuracy of models, which may have important implications in a wide range of areas, from engineering to resource management and decision making.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Capítulo 7 -

7. Presupuesto

En este capítulo, se presentará el presupuesto para la realización de este proyecto. Como se muestra en la Tabla 7.1, se detalla el desglose del presupuesto, donde se especifican los diferentes conceptos de gastos, su estimación y su justificación estimando un coste por hora de 30€.

Concepto	Horas	Coste (€)
Familiarización con el tema propuesto:	40	1200
Análisis del modelo de simulación	30	900
Trabajo con los datos del modelo de simulación	30	900
Preparación de los datos para la utilización de modelos subrogados de aprendizaje automático	60	1800
Selección y entrenamiento de modelos de regresión	70	2100
Ajuste de los modelos	75	2250
Comparación de resultados	35	1050
Conexión de los modelos con la aplicación web	35	1050
Total	375	11250

Tabla 7.1. Desglose por actividades del presupuesto.

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19

Bibliografía

- [1] Cabrera Hernández, M. A. (n.d.). TFG. Github. Retrieved December 27, 2022, from <https://github.com/marcocabrerahdez/TFG>
- [2] Castilla Rodríguez, I. (n.d.). DIAB+. DIAB+. Retrieved May 15, 2023, from <https://diabetes-pre.iaas.ull.es/>
- [3] Chart.js | Open source HTML5 Charts for your website. (n.d.). Retrieved May 9, 2023, from <https://www.chartjs.org/>
- [4] Coeficiente de determinación. (n.d.). Wikipedia. Retrieved May 14, 2023, from https://es.wikipedia.org/wiki/Coeficiente_de_determinaci%C3%B3n
- [5] Efectividad, seguridad y coste-efectividad de los sistemas de monitorización continua de glucosa intersticial en tiempo real (MCG-TR) para la Diabetes Mellitus tipo 1 y 2. Actualización (Issue 133-20-041-8). (2020, octubre). Informes de Evaluación de Tecnologías Sanitarias (SESCS). https://funcanis.es/wp-content/uploads/2020/11/04_SESCS_2019_Act_SMCG_TR_DEF_NIPO.pdf
- [6] Elshawi, R., Al-Mallah, M. H., & Sakr, S. (2019). On the interpretability of machine learning-based model for predicting hypertension. BMC Medical Informatics and Decision Making, 19(1). 10.1186/s12911-019-0874-0
- [7] Flask Documentation (2.3.x). (n.d.). Flask Documentation (2.3.x). Retrieved May 9, 2023, from <https://flask.palletsprojects.com/en/2.3.x/>
- [8] Géron, A. (2017). Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media.
- [9] Google Docs. (n.d.). Google Docs. Retrieved December 21, 2022, from <https://www.google.com/docs/about/?pli=1>
- [10] Google Drive. (n.d.). Google. Retrieved December 21, 2022, from https://www.google.com/intl/es_es/drive/

- [11] López Bastida, J., Oliva, J., Antoñanzas, F., García-Altés, A., Gisbert, R., Mar, J., & Puig-Junoy, J. (2010). Propuesta de guía para la evaluación económica aplicada a las tecnologías sanitarias. Gaceta sanitaria, 24(2), 154-170. 10.1016/j.gaceta.2009.07.011
- [12] Matplotlib. (n.d.). Matplotlib — Visualization with Python. Retrieved December 21, 2022, from <https://matplotlib.org/>
- [13] Mean absolute percentage error. (n.d.). Wikipedia. Retrieved May 14, 2023, from https://en.wikipedia.org/wiki/Mean_absolute_percentage_error
- [14] Misra, R., Shyamasundar, R. K., & Omer, R. (2021). Machine Learning and Big Data Analytics (Proceedings of International Conference on Machine Learning and Big Data Analytics (ICMLBDA) 2021) (A. Chaturvedi, Ed.). Springer Nature.
- [15] Mt hood diabetes challenge network. (n.d.). Mt hood diabetes challenge network. Retrieved December 21, 2022, from <https://www.mthooddiabeteschallenge.com/>
- [16] Next.js by Vercel - The React Framework for the Web. (n.d.). Retrieved May 9, 2023, from <https://nextjs.org/>
- [17] Oliphant, T. E. (n.d.). numpy · PyPI. PyPI. Retrieved December 21, 2022, from <https://pypi.org/project/numpy/>
- [18] Organización Mundial de la Salud. (2022, September 16). Diabetes. Diabetes. Retrieved December 21, 2022, from <https://www.who.int/es/news-room/fact-sheets/detail/diabetes>
- [19] Pandas. (n.d.). pandas - Python Data Analysis Library. Retrieved December 21, 2022, from <https://pandas.pydata.org/>
- [20] Python Release Python 3.11.0. (2022, October 24). Python.org. Retrieved December 21, 2022, from <https://www.python.org/downloads/release/python-3110/>

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- [21] scikit-learn. (n.d.). scikit-learn: machine learning in Python — scikit-learn 1.2.0 documentation. Retrieved December 21, 2022, from <https://scikit-learn.org/stable/>
- [22] SciPy. (n.d.). SciPy.org. Retrieved April 5, 2023, from <https://scipy.org/>
- [23] sklearn.ensemble.RandomForestRegressor — scikit-learn 1.2.0 documentation. (n.d.). Scikit-learn. Retrieved December 27, 2022, from <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>
- [24] sklearn.linear_model.LinearRegression — scikit-learn 1.2.0 documentation. (n.d.). Scikit-learn. Retrieved December 27, 2022, from https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html
- [25] sklearn.neural_network.MLPRegressor — scikit-learn 1.2.2 documentation. (n.d.). Scikit-learn. Retrieved May 7, 2023, from https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html
- [26] sklearn.tree.DecisionTreeRegressor — scikit-learn 1.2.2 documentation. (n.d.). Scikit-learn. Retrieved May 7, 2023, from <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html>
- [27] Stoean, R. (2020). Analysis on the potential of an EA–surrogate modelling tandem for deep learning parametrization: an example for cancer classification from medical images. *Neural Computing and Applications*, 32(2), 313-322. <https://doi.org/10.1007/s00521-018-3709-5>
- [28] Visual Studio Code. (n.d.). Visual Studio Code - Code Editing. Redefined. Retrieved December 21, 2022, from <https://code.visualstudio.com/>

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

- Anexos -

Apéndice A

En este apéndice se muestra el código de la clase *AutoML* que fue explicada en la sección 2.3 de este documento y que se encarga del proceso de entrenamiento y evaluación de los modelos de aprendizaje automático subrogados:

```
"""Module that contains the AutoML class.

This class automates the training and evaluating process of
Machine Learning models.
"""

import importlib
import os

import joblib
import pandas as pd
from sklearn.model_selection import GridSearchCV,
train_test_split
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler

import settings as st
import utils as ut

class AutoML:
    """Class that automates the training and evaluating
    process of Machine Learning models.

    Attributes:
        _name (str): Name of the model.
        _class_name (str): Name of the Machine Learning class.
        _model (str): Machine Learning model.
        _type (str): Type of Machine Learning model.
        _params (hash): Parameters of the model.
        _X_train (pd.DataFrame): Training data.
```

51

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```

    _X_test (pd.DataFrame): Testing data.
    _y_train (pd.DataFrame): Training labels.
    _y_test (pd.DataFrame): Testing labels.
    _y_pred (pd.DataFrame): Predicted labels.
    _model_list_names (List[str]): List of names of the
models.
    _trained_data_names (List[str]): List of names of
trained data.
    """
    __slots__ = [
        '_name',
        '_class_name',
        '_model',
        '_type',
        '_params',
        '_X_train',
        '_X_test',
        '_y_train',
        '_y_test',
        '_y_pred',
        '_model_list_names',
        '_trained_data_names'
    ]

    def __init__(
        self,
        name: str,
        class_name,
        model,
        type_model: str,
        params,
        trained_data_names=[],
        columns_X: pd.DataFrame = pd.DataFrame(),
        columns_Y: pd.DataFrame = pd.DataFrame()) -> None:
        """Constructor for the class.

        Args:
            name (str): Name of the model.
            class_name (str): Name of the Machine Learning
class.
            model (str): Machine Learning model.

```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Arnay del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```

        type_model (str): Type of Machine Learning model.
        params (hash): Parameters of the model.
        trained_data_names (List[str]): List of names of
trained data.
        columns_X (pd.DataFrame): Input columns.
        columns_y (pd.DataFrame): Output columns.

    Returns:
        None
    """
    # Initialize the attributes
    self._name = name
    self._model_list_names = model
    self._class_name =
[importlib.import_module(class_name[i])
    for i in range(len(class_name))]

    self._type = type_model
    self._params = params
    self._trained_data_names = trained_data_names
    self._y_pred = None

    # Create the model
    self._model = []
    for i in range(len(model)):
        if (model[i] == 'LinearRegression'):

self._model.append(getattr(self._class_name[i], model[i]))
        else:

self._model.append(getattr(self._class_name[i], model[i])(
    random_state=st.RANDOM_STATE))
        if self._type == 'single':
            self._X_train, self._X_test, self._y_train,
self._y_test = train_test_split(
                columns_X, columns_Y, test_size=st.TEST_SIZE,
random_state=st.RANDOM_STATE)

            elif self._type == 'multiple' or self._type ==
'global':
                self._X_train, self._X_test, self._y_train,

```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```

self._y_test = ut.get_splitted_data(
    self._trained_data_names, self._type)
    else:
        raise Exception('El tipo de modelo no es válido.')

    # Save the data use by single models to be able to use
    them in multiple
    # and global models
    ut.save_splitted_data(
        self._X_train,
        self._X_test,
        self._y_train,
        self._y_test,
        self._name,
        self._type)

    # Get the columns of the input data
    columns = self._X_train.columns.copy()

    # Preprocces the data using the StandardScaler to
    normalize the data
    scaler = StandardScaler()
    scaler.fit(self._X_train)
    self._X_train = scaler.transform(self._X_train)
    self._X_test = scaler.transform(self._X_test)

    # Assign the correct data type to the data to allow
    the use of the data
    # in the models
    self._X_train = pd.DataFrame(
        self._X_train, columns=columns)
    self._X_test = pd.DataFrame(
        self._X_test, columns=columns)

def train(self) -> None:
    """Train the models.

    Returns:
        None
    """
    print('Training the models...')
  
```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```

    # Create a pipeline with the model and the parameters
    to be used in the
    # grid search
    pipe = [Pipeline(['model', self._model[i]])
             for i in range(len(self._model))]

    # Create a grid search with the pipeline and the
    parameters
    # Grid search is used to find the best parameters for
    the model
    grid_search = [
        GridSearchCV(
            pipe[i],
            param_grid=self._params[i],
            cv=5,
            refit=True,
            scoring='r2') for i in range(
            len(pipe))]

    # Fit the grid search with the training data
    for i in range(len(grid_search)):
        grid_search[i].fit(self._X_train, self._y_train)
        self._model[i] = grid_search[i].best_estimator_

def predict(self) -> None:
    """Predict the output data.

    Returns:
        None
    """
    print('Predicting the data...')
    # The predict method returns a list of lists,
    # so the data will be converted in
    save_predictions_results
    self._y_pred = [self._model[i].predict(self._X_test)
                     for i in range(len(self._model))]

def _save_predictions_results(self, folder: str = '') ->
None:
    """Save the predictions results in a .xlsx file.

```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```

        Args:
            folder (str): Name of the folder where the results
will be saved.

        Raises:
            ValueError: If the type of model is not valid.

        Returns:
            None

        """
        if self._type not in ['single', 'multiple', 'global']:
            raise ValueError('El tipo de modelo no es
válido.')

        for i, y_pred_i in enumerate(self._y_pred):
            model_dir = ''
            if self._type == 'single':
                model_dir = st.SINGLE_PREDICTIONS_DIR
            elif self._type == 'multiple':
                model_dir = st.MULTIPLE_PREDICTIONS_DIR
            elif self._type == 'global':
                model_dir = st.GLOBAL_PREDICTIONS_DIR
            else:
                raise ValueError('El tipo de modelo no es
válido.')

            if folder != '':
                model_list_dir = os.path.join(
                    model_dir, self._model_list_names[i],
folder)
            else:
                model_list_dir = os.path.join(
                    model_dir, self._model_list_names[i])
            if not os.path.exists(model_list_dir):
                os.makedirs(model_list_dir)

            y_pred_df = pd.DataFrame(y_pred_i,
columns=self._y_test.columns)
            file_path = os.path.join(model_list_dir,
f'{self._name}.xlsx')
    
```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
 Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Arnay del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19


```

y_pred_df.to_excel(file_path, index=False)

def _save_model(self, folder: str = '') -> None:
    """Save the model in a .pkl file.

    Args:
        folder (str): Name of the folder where the model
will be saved.

    Raises:
        ValueError: If the type of model is not valid.

    Returns
        None
    """
    model_path = []
    model_dir = ''
    if self._type == 'single':
        model_dir = st.SINGLE_MODEL_DIR
    elif self._type == 'multiple':
        model_dir = st.MULTIPLE_MODEL_DIR
    elif self._type == 'global':
        model_dir = st.GLOBAL_MODEL_DIR
    else:
        raise ValueError('El tipo de modelo no es
válido.')

    for i, model in enumerate(self._model):
        model_name = f'{self._name}.pkl'
        if folder != '':
            model_list_dir = os.path.join(
                model_dir, self._model_list_names[i],
folder)
        else:
            model_list_dir = os.path.join(
                model_dir, self._model_list_names[i])
        model_path.append(os.path.join(model_list_dir,
model_name))
        if not os.path.exists(model_list_dir):
            os.makedirs(model_list_dir)
        joblib.dump(model, model_path[i])

```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YyJY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ UNIVERSIDAD DE LA LAGUNA	Fecha 22/05/2023 15:53:45
Iván Castilla Rodríguez UNIVERSIDAD DE LA LAGUNA	22/05/2023 16:07:32
Rafael Aray del Arco UNIVERSIDAD DE LA LAGUNA	22/05/2023 17:06:19

```
def save(self, folder: str = '') -> None:
    """Save the model and the predictions results.

    Args:
        folder (str): Name of the folder where the model
and the results.

    Returns
        None
    """
    print('Saving the results...')
    self._save_model(folder)
    self._save_predictions_results(folder)
```

Este documento incorpora firma electrónica, y es copia auténtica de un documento electrónico archivado por la ULL según la Ley 39/2015.
Su autenticidad puede ser contrastada en la siguiente dirección <https://sede.ull.es/validacion/>

Identificador del documento: 5429785 Código de verificación: nPGy9YjY

Firmado por: MARCO ANTONIO CABRERA HERNÁNDEZ
UNIVERSIDAD DE LA LAGUNA

Fecha 22/05/2023 15:53:45

Iván Castilla Rodríguez
UNIVERSIDAD DE LA LAGUNA

22/05/2023 16:07:32

Rafael Aray del Arco
UNIVERSIDAD DE LA LAGUNA

22/05/2023 17:06:19