

Predicción diabetes tipo 2 a partir de un cuestionario telefónico

11 de febrero de 2026

Título del trabajo	Predicción diabetes tipo 2 a partir de un cuestionario telefónico
Nombre del autor:	Carlos Alejandro Pérez Casares
Fecha de entrega:	5/1/2026
Área de trabajo final:	Área de la salud
Idioma del trabajo:	Castellano
Licencia de la memoria:	Creative Commons BY-NC
Palabras clave:	Prediction models, Machine learning, Type 2 diabetes mellitus, BRFSS dataset

Resumen

La diabetes mellitus tipo 2 (DM2) es una enfermedad crónica de gran difusión cuyo diagnóstico tardío aumenta complicaciones y costes sanitarios. Los métodos clínicos tradicionales de cribado pueden resultar costosos o poco accesibles para la población general. El objetivo de este trabajo es diseñar un modelo predictivos que permitan identificar de manera temprana el riesgo de DM2 a partir de datos obtenidos mediante encuestas telefónicas de salud pública, concretamente el conjunto de datos BRFSS 2014 del CDC (EEUU). El proyecto emplea una metodología basada en el proceso de minería de datos y aprendizaje automático. Tras limpiar el conjunto de datos original, se transformaron, después se mitigaron sesgos, se realizó la reducción de la dimensionalidad del conjunto de datos mediante Recursive Feature Elimination (RFE) permitiendo la selección de las variables más predictivas, se equilibró la variable objetivo mediante técnicas de over-sampling (SMOTENC) para corregir el desequilibrio de clases. Se entrenó y validó los modelos de clasificación supervisada Regresión Logística y CatBoost, evaluando su rendimiento con métricas como precisión, sensibilidad, especificidad y AUC-ROC. Al modelo CatBoost se le aplicó SHAP, que es un método para explicar las predicciones del modelo. Finalmente se integró el modelo calculado con CatBoost en una aplicación con interfaz gráfica para hacer un cribado a partir de cinco variables más explicativas calculadas con SHAP.

Como conclusión, el riesgo de diabetes depende tanto de parámetros clínicos como de hábitos de vida, acceso a la atención médica y factores demográficos.

Abstract

Type 2 diabetes mellitus (T2DM) is a widespread chronic disease whose late diagnosis increases complications and health costs. Traditional clinical screening methods can be expensive or inaccessible to the general population. The objective of this work is to design predictive models that allow early identification of the risk of DM2 from data obtained through public health telephone surveys, specifically the BRFSS 2014 data set from the CDC (USA). The project uses a methodology based on the data mining and machine learning process. After cleaning the original data set, they were transformed, then biases were mitigated, the dimensionality reduction of the data set was carried out using Recursive Feature Elimination (RFE) allowing the selection of the most predictive variables, the target variable was balanced using over-sampling techniques (SMOTENC) to correct the class imbalance. The Logistic Regression and CatBoost supervised classification models were trained and validated, evaluating their performance with metrics such as precision, sensitivity, specificity and AUC-ROC. SHAP was applied to the CatBoost model, which is a method to explain the model predictions. Finally, the model calculated with CatBoost was integrated into an application with a graphical interface to perform a screening based on five more explanatory variables calculated with SHAP.

In conclusion, the risk of diabetes depends on both clinical parameters

and lifestyle habits, access to medical care and demographic factors.

Índice general

1. Introducción	5
1.1. Contexto y justificación del Trabajo	5
1.2. Objetivos del Trabajo	7
1.3. Impacto en sostenibilidad, ético-social y de diversidad	8
1.4. Enfoque y método seguido	9
1.5. Breve resumen de productos obtenidos	10
1.6. Breve descripción de los otros capítulos de la memoria	10
2. Materiales y métodos	12
2.1. Fuentes de datos utilizadas	12
2.2. Herramientas tecnológicas	12
2.3. Uso combinado de Copilot y O'Reilly en el notebook	13
2.4. Procesamiento de datos	14
2.4.1. Limpiar el conjunto de datos	14
2.4.2. Transformación de los datos	17
2.4.3. Mitigación de sesgos con AIF360	17
2.4.4. Seleccionar las variables más predictivas	19
2.4.5. Balanceo con SMOTENC	20

2.4.6. Entrenamiento de modelos	21
2.4.7. Cómo se entrena y valida Regresión Logística	23
2.4.8. Cómo se entrena y valida CatBoost	24
2.4.9. Evaluar el desempeño de los modelos	25
2.4.10. SHAP	27
2.4.11. Relación entre RFE y SHAP	28
2.5. Análisis de riesgos	30
3. Resultados	32
3.1. Procesamiento de datos	32
3.1.1. Mitigación de sesgos con AIF360	32
3.1.2. Reducción de variables	36
3.1.3. Balancear la variable objetivo	39
3.1.4. Explicabilidad con SHAP	39
3.2. Estado del arte	41
4. Conclusiones y trabajos futuros	44
5. Glosario	46
6. Bibliografía	48
7. Anexos	53

Índice de figuras

2.1. Diagrama de flujo de las etapas de creación de modelos y su evaluación	15
2.2. Conjunto de datos desbalanceado para DIABETE3	16
2.3. Valores en la variable DIABETE3 del juego de datos original .	16
2.4. Variable edad (_AGEG5YR)	17
3.1. Antes y después de la mitigación	35
3.2. Balanceo mediante oversampling con SMOTENC	36
3.3. Gráfica con las variables más explicativas y su grado de importancia	37
3.4. Diagrama SHAP con XGBoost	40
7.1. Aplicación de cribado con interfaz gráfica	54

Índice de cuadros

2.1. Matriz de confusión	25
3.1. Sesgo según el género	33
3.2. Sesgo según edad	33
3.3. Interpretación métrica Disparate Impact de AIF360	34
3.4. Métricas antes y después de mitigar sesgo por edad	35
3.5. Las veinte variables más explicativas según RFE	36
3.6. Las veinte variables más explicativas según RFE	38
3.7. Variables más influyentes según SHAP	39
3.8. Etapas de mi TFG	41

Capítulo 1

Introducción

1.1. Contexto y justificación del Trabajo

El ámbito de este trabajo es el de la salud. Un proyecto donde se diseña e implementa un sistema de detección precoz de una enfermedad: la diabetes tipo 2.

Este proyecto tiene un impacto social positivo y es una contribución al ámbito de la salud pública. El objetivo es crear un modelo de clasificación preciso a partir de datos obtenidos de forma legal y transparente mediante encuestas telefónicas oficiales, lo que facilitaría un cribado preventivo más asequible y temprano para los pacientes, reduciendo costes en el sistema de salud comparado con análisis clínicos de laboratorio.

La diabetes tipo 2 (DM2) es un estado en el que la insulina producida no puede mantener estable el nivel de glucosa en sangre en todo el cuerpo, y se presenta con mayor frecuencia en personas mayores de 40 años. El tipo más común de diabetes es la DM2, que representa entre el 90 y el 95 % de todos

los casos de diabetes diagnosticados en todo el mundo. (Ashisha, 2024)

Además es una enfermedad crónica de gran difusión cuyo diagnóstico tardío aumenta complicaciones y costes sanitarios. Los métodos clínicos tradicionales de cribado pueden resultar costosos o poco accesibles para la población general. El objetivo de este trabajo es diseñar modelos predictivos que permitan identificar de manera temprana el riesgo de DM2 a partir de datos obtenidos mediante encuestas telefónicas de salud pública, concretamente el conjunto de datos BRFSS 2014 del CDC.

El BRFSS es una herramienta epidemiológica diseñada para medir la distribución y los determinantes de los principales factores de riesgo de enfermedades crónicas en la población adulta de Estados Unidos. Su estructura, basada en variables demográficas, conductuales, clínicas y sociales, permite aplicar principios epidemiológicos clásicos —como el análisis de distribución, determinantes y prevalencia— al estudio de la diabetes tipo 2. En este trabajo, estos conceptos se integran en un enfoque de aprendizaje automático (Moccia, 2024), donde los modelos predictivos utilizan precisamente los factores epidemiológicos identificados en la literatura (edad, IMC, comorbilidades, determinantes sociales) para estimar el riesgo de diabetes tipo 2. La interpretabilidad mediante SHAP permite traducir los resultados del modelo a un lenguaje epidemiológico, reforzando la relación entre la teoría epidemiológica y la predicción por ordenador.

1.2. Objetivos del Trabajo

I. **Objetivos generales:** Diseñar e implementar un modelo predictivo para identificar el riesgo de diabetes tipo 2 basado en datos del BRFSS. A partir de este modelo crear una aplicación para cribado especificando solamente cinco variables.

II. **Objetivos específicos:**

- Limpiar el conjunto de datos del BRFSS eliminando filas y columnas innecesarias y validando la integridad de los registros (CDC, 2014).
- Balancear la variable objetivo mediante técnicas de over-sampling adaptadas al desequilibrio de clases (Lemaître:n.d.).
- Imputar los valores faltantes utilizando el algoritmo SimpleImputer de sklearn.
- Seleccionar las variables más predictivas mediante Recursive Feature Elimination (Scikit-learn (s.f.)).
- Mitigación de sesgos mediante AIF360.
- Entrenar y optimizar un algoritmos de aprendizaje automático (UOC Labs).
- Evaluar el desempeño de los modelos con métricas de precisión, sensibilidad, especificidad y AUC-ROC (Saito y Rehmsmeier: 2015).
- Comparar los resultados de los modelos para identificar el algoritmo con mayor precisión predictiva.

- Aplicar SHAP para interpretabilidad. (Allani, 2025; Awan, 2024)
- Integrar el modelo seleccionado en una aplicación GUI desarrollada en Python/Tkinter para generar predicciones individuales (Ramírez: 2022).

1.3. Impacto en sostenibilidad, ético-social y de diversidad

Sobre el impacto ético ha de decirse que la protección de la privacidad se garantiza mediante el uso de registros des identificados y la exclusión de datos personales sensibles.

Sin embargo, al basarse en encuestas telefónicas, se excluyen poblaciones vulnerables, lo que genera un sesgo en la selección de la muestra y afecta la equidad y validez del modelo (impacto de diversidad). Las poblaciones más vulnerables -como personas mayores sin acceso a tecnología, migrantes con barreras lingüísticas o individuos en situación de exclusión social- pueden quedar fuera del proceso de recogida de datos. Lo que genera un sesgo en la selección de la muestra y afecta la equidad y validez de los modelos. Perpetuándose, así, desigualdades en el acceso a la prevención y al diagnóstico.

La detección precoz de la diabetes tipo 2 tiene un impacto directo en la calidad de vida de millones de personas (impacto social). Al identificar a tiempo a los individuos con riesgo elevado, es posible aplicar cambios en el estilo de vida y tratamientos farmacológicos que retrasen o eviten la aparición de complicaciones graves (retinopatía, neuropatía, enfermedades cardiovas-

culares) (Knowler: 2002).

La detección precoz de la diabetes tipo 2 tiene las siguientes ventajas entre otras:

- Disminución de costes sanitarios: prevenir complicaciones reduce ingresos hospitalarios y tratamientos crónicos de alto coste.
- Mejora en pronóstico clínico: los pacientes intervenidos tempranamente presentan menores tasas de mortalidad y menor necesidad de intervenciones invasivas.
- Mejora de la actitud del paciente: una predicción personalizada facilita la adherencia a dietas, ejercicio y control médico regular.

1.4. Enfoque y método seguido

Este proyecto seguirá el proceso habitual de minería de datos para crear modelos predictivos del riesgo de diabetes tipo 2 mediante algoritmos de aprendizaje automático (algoritmos de clasificación binaria supervisada).

Las etapas clave son: Limpieza de los datos, Transformación de los datos, Reducción de dimensionamiento, Construcción de los modelos, Validación de los modelos, Trato de sesgos y la Equidad en los modelos. Finalmente se creará una aplicación gráfica para que mediante los modelos entrenados pueda inferir si un paciente tendrá o no diabetes.

1.5. Breve resumen de productos obtenidos

- notebook.html: notebook Jupyter contiene limpieza, imputación con RandomForestRegressor, balanceo del conjunto de datos y reducción de dimensiones hasta 30 variables o características. Mitigación de sesgos, selección de las 13 variables más predictivas. Entrenamiento de modelos. Métricas de los modelos. Aplicación de SHAP a CatBoost.
- memoria_TFG.pdf: incluye introducción, objetivos, metodología y justificación del cambio de algoritmo.
- Guion xpt_to_csv: guion en Python con el que se ha convertido el archivo en formato XPT (SAS Transport Format) al formato CSV.
- Aplicación con interfaz gráfica para cribado.

1.6. Breve descripción de los otros capítulos de la memoria

Capítulo 2 Materiales y métodos

2.1. Fuentes de datos utilizadas

2.2. Herramientas tecnológicas

2.3. Creación del notebook

2.4. Procesamiento de datos con las diferentes etapas en las que he dividido el trabajo.

2.4.3. Cómo se abordan los sesgos y la equidad en los modelos. Detección y mitigación de sesgos.

2.4.5. Desbalance de clases, limitaciones del BRFSS, problemas de generalización.

Un modelo entrenado en datos del BRFSS puede no generalizar bien a otras poblaciones con características socioeconómicas, culturales o sanitarias diferentes.

2.5. Análisis de riesgos

Capítulo 3 Resultados

Resultados de los entrenamientos de modelos .

Métricas de los modelos creados.

Elección del modelo con mejor rendimiento.

Aplicación de SHAP al mejor modelo.

Pasos y técnicas aplicados al conjunto de datos BRFSS según la literatura científica más reciente (años 2024, 2025).

Capítulo 4 Conclusiones y trabajos futuros

Capítulo 2

Materiales y métodos

2.1. Fuentes de datos utilizadas

En este trabajo se utiliza un conjunto de datos público que está en

https://www.cdc.gov/brfss/annual_data/2014/files/LLCP2014XPT.ZIP.

Este, es una recopilación de datos de personas a quienes se han llamado por teléfono por un servicio de salud que funciona en EE.UU (Behavioral Risk Factor Surveillance System). Los datos se recopilan presentando un cuestionario llamando a teléfonos fijos y móviles.

“Es un sistema de encuestas telefónicas relacionadas con la salud que recopilan datos de todos los estados sobre los residentes de EE.UU. con respecto a sus conductas de riesgo relacionadas con la salud, condiciones de salud crónicas y uso de servicios preventivos.”(CDC, 2014)

2.2. Herramientas tecnológicas

El conjunto de datos LLCP2014.XPT está en formato nativo de la aplicación de tratamiento estadístico SAS. Este se ha convertido a formato csv

mediante un guion creado ad-hoc en Python.

Se ha empleado el lenguaje de programación Python, librerías como Pandas, scikit-learn y otras relacionadas con el aprendizaje automático, además del uso de notebooks Jupyter. La imputación de los valores faltantes del dataframe y el entrenamiento de todos los modelos se ejecutaron en CPU.

2.3. Uso combinado de Copilot y O'Reilly en el notebook

Durante el proceso de programación en el notebook con Python se emplearon dos recursos: Microsoft Copilot y la biblioteca O'Reilly. Usé Copilot como asistente interactivo y búsqueda de referencias bibliográficas, mientras que O'Reilly también proporcionó referencias bibliográficas y ejemplos prácticos de libros.

El empleo de Copilot y O'Reilly se limitó a funciones de apoyo, manteniendo siempre mi criterio para comprobar, adaptar y documentar cada sugerencia. Se prestó atención a la justificación de cada componente del código, donde la autoría y responsabilidad intelectual es del mía.

Es importante resaltar que el uso de Copilot para sugerir código no es directo. Comete muchos fallos por lo que hay que supervisar el código que muestra de manera iterativa.

2.4. Procesamiento de datos

Este proyecto sigue el proceso tradicional de minería de datos para crear modelos predictivos del riesgo de diabetes tipo 2 mediante algoritmos de aprendizaje automático. En la Figura 2.1 se ven los pasos generales seguidos en este TFG.

A continuación muestro las etapas que he seguido.

2.4.1. Limpiar el conjunto de datos

Limpiar el conjunto de datos del BRFSS eliminando filas y columnas innecesarias y validando la integridad de los registros (CDC, 2014).

Las columnas ["_STATE", "FMONTH", "IDATE", "IMONTH", "IDAY", "IYEAR", "DISPCODE", "SE QNO", "_PSU"] se refieren a fechas y las elimino del conjunto de datos porque no son útiles para el propósito de este TFG ya que me centro en analizar los datos de solo un año, el 2014.

DIABETE3 es la variable objetivo. En la Figura 2.3 se muestra la codificación de sus valores. Sólo nos quedamos con las filas con valores 1: Sí diabetes y 3: No diabetes que luego mapeo a Clase 1 (sí diabetes), Clase 0 (no diabetes) (Figura 2.2).

Variable objetivo (dependiente) : DIABETE3. Clase 0 (no diabetes): 348061 filas; Clase 1 (sí diabetes): 60538 filas. El conjunto de datos está muy desequilibrado. Luego lo corregiré balanceando con SMOTENC (ver apartado 2.4.5).

Elimino la variable __AGEG5YR (Figura 2.4) que son menores de 34 años. Los menores de 34 años tienen diabetes de tipo 1 y no diabetes de tipo 2

Figura 2.1: Diagrama de flujo de las etapas de creación de modelos y su evaluación

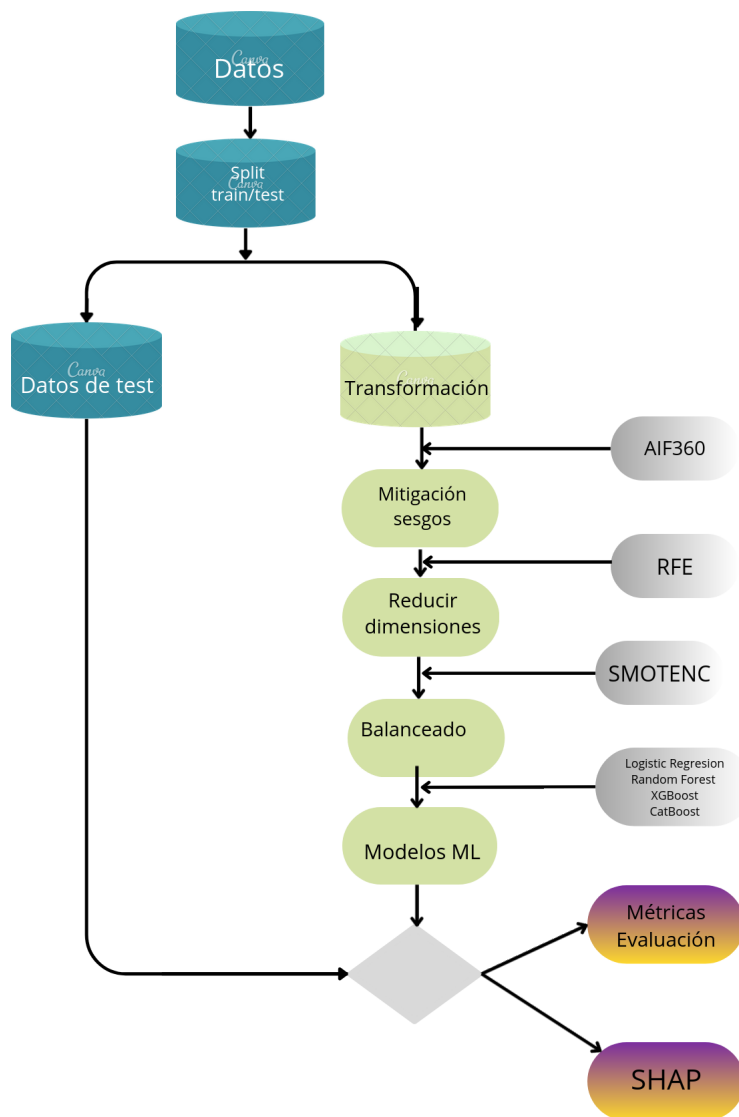


Figura 2.2: Conjunto de datos desbalanceado para DIABETE3

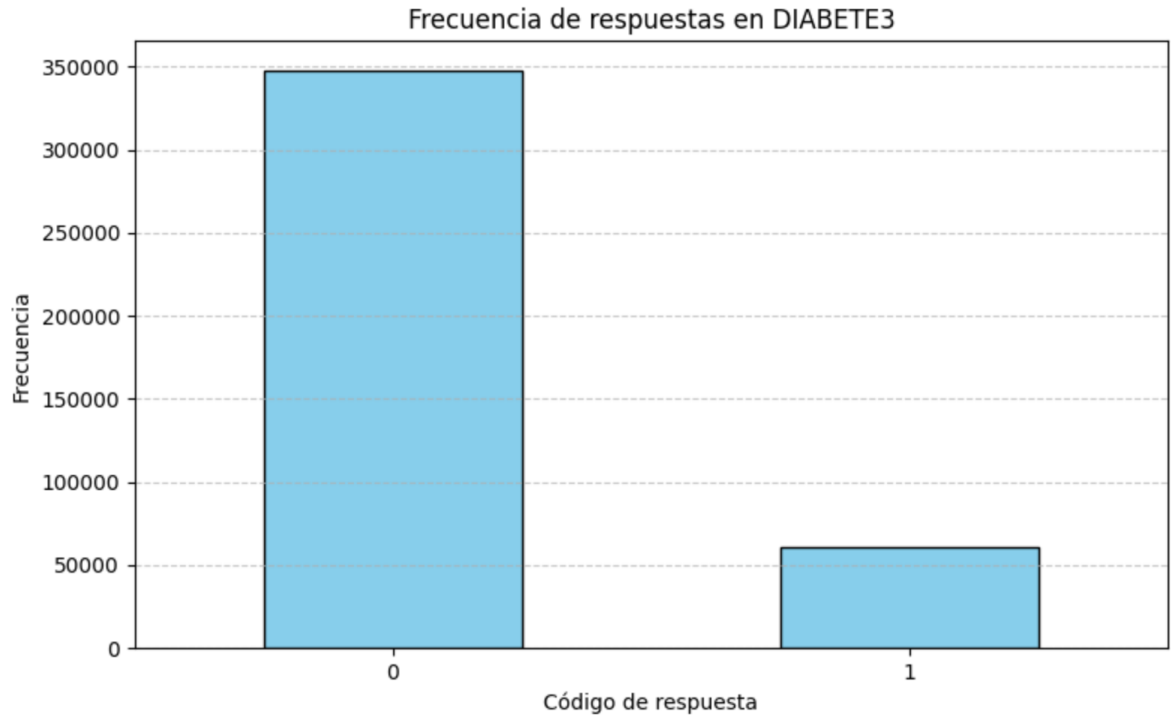


Figura 2.3: Valores en la variable DIABETE3 del juego de datos original

Ever told) you have diabetes

Section: 6.12 Chronic Health Conditions

Column: 105

Prologue:

Description: (Ever told) you have diabetes (If "Yes" and respondent is female, ask "Was this only when you were pregnant?". If Respondent says pre-diabetes or borderline diabetes, use response code 4.)

Type: Num

SAS Variable Name: DIABETE3

Value	Value Label	Frequency	Percentage	Weighted Percentage
1	Yes	61,118	13.15	10.53
2	Yes, but female told only during pregnancy—Go to Section 07.7.1 LASTDEN3	4,207	0.91	1.01
3	No—Go to Section 07.7.1 LASTDEN3	390,827	84.11	86.78
4	No, pre-diabetes or borderline diabetes—Go to Section 07.7.1 LASTDEN3	7,668	1.65	1.48
7	Don't know/Not Sure—Go to Section 07.7.1 LASTDEN3	551	0.12	0.14
9	Refused—Go to Section 07.7.1 LASTDEN3	291	0.06	0.05
BLANK	Not asked or Missing	2		

Figura 2.4: Variable edad (_AGEG5YR)

Reported age in five-year age categories calculated variable

Calculated Variables: 8.11 Calculated Variables Type: Num

Column: 2232-2233 SAS Variable Name: _AGEG5YR

Prologue:

Description: Fourteen-level age category

Value	Value Label	Frequency	Percentage	Weighted Percentage
1	Age 18 to 24 Notes: 18 <= AGE <= 24	24,198	5.21	12.92
2	Age 25 to 29 Notes: 25 <= AGE <= 29	19,891	4.28	8.30
3	Age 30 to 34 Notes: 30 <= AGE <= 34	23,662	5.09	8.97

(Akil, 2021).

2.4.2. Transformación de los datos

División del conjunto de datos en dos particiones: entrenar/prueba

Imputación de valores faltantes (en train)

Codificación (en train). Convierte categorías en números enteros.

Escalado (en train). Convertir las variables a una escala con: media = 0, desviación estándar = 1. Para que los modelos no se vean afectados por variables con escalas distintas, aunque en mi caso solo es útil el modelado con la Regresión logística porque a los modelos creados con Random Forest, XGBoost, CatBoost no les afecta pero tampoco los perjudica.

2.4.3. Mitigación de sesgos con AIF360

Para la detección y mitigación de sesgos he usado AIFairness360 (Bellamy, 2018) combinándolo con el entrenamiento de modelo Regresión logística para comprobar si los sesgos se mitigan después de aplicar las correcciones al modelo (AI Fairness 360, (n.d.)).

Flujo: Transformar datos → métricas AIF360(antes) → entrenar modelo → métricas AIF360 (después).

Hay que aplicar AIF360 antes de balancear con SMOTENC para que no haya fuga de información (data leakage).

Hay que mapear la columna DIABETE3 a una etiqueta binaria para que AIF360 y los clasificadores la interpreten correctamente.

- Sí diabetes: 1 → debe convertirse en 1
- No diabetes: 3 → debe convertirse en 0

Pasos pendientes. Flujo: **Preprocesamiento → Mitigación → Reducción de características → Creación de modelos.**

La función `DISPARATE_IMPACT()` de AIF360 calcula la división entre resultado favorable del grupo no privilegiado y del grupo privilegiado.

El Disparate Impact es el ratio de resultados positivos entre grupos. El Disparate Impact ideal es el que está cercano a 1. Fórmula:

$$\text{Disparate Impact} = \frac{P(\text{resultado positivo} \mid \text{grupo protegido})}{P(\text{resultado positivo} \mid \text{grupo privilegiado})}$$

Con AIF360, al detectar `BINARYLABELDATASETMETRIC` con sesgo por `AGE_GROUP_BIN` edad(usando como atributo protegido), podemos aplicar algoritmos de mitigación de sesgo que generan un nuevo conjunto de datos transformado. Ese dataset se puede exportar a CSV.

Flujo: Baseline → Mitigación con Reweighing → Post-mitigación

El clasificador usado en las métricas ha sido LogisticRegression y se ha

mitigado el sesgo con **Reweighting** (ajuste de pesos)..

2.4.4. Seleccionar las variables más predictivas

Seleccionar las variables más predictivas mediante el método

`sklearn.feature_selection.RFE`. RFE (Eliminación Recursiva de Características) es una técnica de selección de variables que busca identificar el subconjunto más relevante de características --es decir, variables o columnas del juego de datos-- para un modelo predictivo. Pertenecce a la categoría de métodos de envoltorio (wrapper)(GeeksforGeeks, 2023), porque utiliza un modelo de aprendizaje para evaluar la importancia de cada variable. RFE con Regresión Logística. Este pipeline aplica RFE usando un Regresión Logística como modelo base, para seleccionar las 50 variables más relevantes del conjunto de datos.

Su funcionamiento es el siguiente:

1. **Entrenamiento inicial.** Se ajusta el modelo con Regresión Logística usando todas las variables.
2. **Evaluación de importancia.** Se calcula la importancia de cada variable.
3. **Eliminación de variables menos relevantes.** Se eliminan las variables menos importantes.
4. **Repetición del proceso.** Se vuelve a entrenar el modelo con las variables restantes y se repite el proceso hasta alcanzar el número deseado de variables (`N_FEATURES_TO_SELECT`).

He reducido primero las dimensiones del conjunto de datos (escogiendo como parámetro 50 variables) para reducir el tiempo de cálculo del balanceado con SMOTENC.

2.4.5. Balanceo con SMOTENC

Balancear la variable objetivo mediante técnicas de over-sampling adaptadas al desequilibrio de clases (Lemaître,(n.d.)).

1. Evitamos sesgos en la selección de variables: si la clase está desbalanceada, los algoritmos de selección de características pueden favorecer las variables que explican mejor la clase mayoritaria, ignorando señales útiles para la clase minoritaria.

2. Mejor representación de patrones: al balancear antes, aseguramos que los patrones de ambas clases estén presentes en el proceso de reducción.

3. Mayor robustez del modelo final: la reducción de variables sobre un conjunto balanceado tiende a preservar variables relevantes para ambas clases, lo que mejora la generalización

He usado la técnica SMOTENC (Mukherjee, 2021) porque esta es una técnica de re-muestreo que genera ejemplos sintéticos para la clase minoritaria en datasets desbalanceados, preservando las variables categóricas. Es una versión de SMOTE adaptada para datos mixtos (numéricos + categóricos). En nuestro caso la clase minoritaria es «diabetes sí».

Aunque todas las columnas sean numéricas, algunas pueden representar categorías codificadas como números (por ejemplo, 1 : masculino, 2 : femenino) como es nuestro caso. SMOTENC permite tratar las columnas como

categorías, aunque estén codificadas numéricamente. Evita promediar valores categóricos, lo cual sería incorrecto (por ejemplo, generar un valor 1.5 entre 1 = masculino y 2 = femenino). Es muy conveniente separar explícitamente las columnas categóricas de las numéricas cuando se usa SMOTENC, incluso si todas están codificadas como números.

En el bloque de código las columnas del dataframe se separan de manera automática, mediante el uso de un umbral, en categóricas y numéricas. (Abhishek, 2021)

2.4.6. Entrenamiento de modelos

En unas primeras versiones del modelo creado con el método Naive Bayes, su evaluación daba predicciones perfectas, lo cual era sospechoso. Al final descubrí que entre las características del conjunto de datos había una variable ("DIABAGE2") que predecía por sí misma el 100 % de la variable objetivo ("DIABETE3"). Descartando "DIABAGE2" del dataset las métricas de evaluación de los modelos presentaban un aspecto más realista. La variable DIABAGE2 contiene información que está directamente relacionada o derivada del objetivo (DIABETE3), lo que permitía al modelo aprender una relación artificialmente perfecta. Este fenómeno se conoce como **feature target leakage** (o fuga de información). Genera métricas engañosamente altas (como precisión perfecta, Accuracy: 1.0), que no se da en datos reales y el modelo no generaliza.

Clasificación es la tarea de predecir cuál de un conjunto de clases (categorías) a las que pertenece un ejemplo. En nuestro caso son dos clases;

clasificación binaria (diabetes sí o no). {1: Sí diabetes; 0: No diabetes}

Regresión Logística para calcular variables SHAP

La regresión logística es uno de los modelos más adecuados y ampliamente utilizados para problemas de clasificación binaria. Se trata de un modelo de clasificación que estima la probabilidad de pertenencia a una clase. Se adapta bien a problemas binarios (sí/no). Ofrece probabilidades claras que facilitan decisiones médicas. Es interpretable: cada variable muestra su impacto en el riesgo. Tiene uso extendido en epidemiología. Maneja bien variables categóricas y continuas. Es robusta incluso con clases des balanceadas. Cuenta con amplia validación científica en estudios de diabetes tipo 2. Este modelo es adecuado cuando queremos predecir una variable Y a partir de otras variables (variables X) que formarían una combinación lineal de variables. (Martínez, 2014)

CatBoost (Categorical Boosting) para crear modelo de cribado

Se entrenan un modelo de clasificación binaria de manera supervisada (UOC Labs) para crear la aplicación python de cribado. El método escogido para la clasificación binaria es »CatBoost«.

CatBoost es una nueva biblioteca de gradient boosting. Supera a las implementaciones más avanzadas de árboles de decisión utilizando gradient boosting, como XGBoost. Es un algoritmo basado en árboles de decisión. CatBoost destaca por su sensibilidad a los hiperparámetros y la importancia de su ajuste. CatBoost es ideal para datos heterogéneos. (Hancock, 2020) Nuestro dataset es heterogéneo porque mezcla variables continuas (peso, edad, IMC), variables categóricas codificadas como números (empleo, salud

percibida, chequeos), variables ordinales/discretas (visitas médicas, días de consumo de alcohol) y la variable binaria objetivo (diabetes sí/no). Por lo tanto, es adecuado para aplicar CatBoost. Además permite la interpretación de las variables con SHAP.

2.4.7. Cómo se entrena y valida Regresión Logística

Preparación de datos:

- Se lee el CSV con las variables seleccionadas. El csv de entrada está balanceado. Se reemplazan los valores de “No sabe/No responde” (88, 77, 99, etc.) por NaN para tratarlos como nulos. Se definen las variables explicativas (FEATURES) y la variable objetivo (TARGET). Balanceo de clases. Se usa `train_test_split` con `stratify=y` para mantener la proporción de clases en train y test. Pipeline: preprocesamiento + modelo de clasificación. Imputación: rellena valores nulos con `SimpleImputer(strategy="most_frequent")`. Codificación: convierte variables categóricas en variables binarias con `OneHotEncoder`.

Definición del modelo:

- Regresión logística: se entrena con `class_weight="balanced"` para compensar posibles des balances, utilizando el solver `saga` (eficiente para datos grandes y variables categóricas).

Entrenamiento:

- Todo se integra en un pipeline para que el preprocesamiento y el modelo se apliquen juntos. Validación cruzada estratificada (5 splits). Se

utiliza StratifiedKFold para dividir los datos en 5 particiones manteniendo la proporción de clases. Se entrena el modelo en el conjunto de entrenamiento. Se predice en el conjunto de validación.

Evaluación del modelo:

- Se calculan métricas: Accuracy, Precision, Recall, F1 y ROC-AUC

2.4.8. Cómo se entrena y valida CatBoost

Datos de entrada

- Selección de variables explicativas (X) y la variable objetivo (y).
- Especificación explícita de la lista de variables categóricas.
- División en train/test o empleo de validación cruzada estratificada.
- Definición del modelo.
- Se utiliza CatBoostClassifier de la librería catboost.

Parámetros

iterations: número de árboles (ej. 500); *depth*: profundidad máxima de cada árbol; *learning_rate*: tasa de aprendizaje; *loss_function*="Logloss" para clasificación binaria; *eval_metric*="AUC" para evaluar rendimiento; *class_weights* si hay desbalance de clases; *random_seed* para reproducibilidad.

Entrenamiento

Se ajusta el modelo con *fit*(X_train, y_train, *cat_features*=lista_categoricas, *eval_set*=(X_test, y_test)).

Cuadro 2.1: Matriz de confusión

	Predicción: No diabetes	Predicción: Sí diabetes
Real: No diabetes	TN	FP
Real: Sí diabetes	FN	TP

Evaluación del modelo:

Se calculan las métricas Accuracy, Precision , Recall, F1 y ROC_AUC.

2.4.9. Evaluar el desempeño de los modelos

Evaluar el desempeño de los modelos con métricas de precisión, sensibilidad, especificidad y AUC-ROC (Saito y Rehmsmeier, 2015).

El F1- score es una forma práctica de combinar precisión y sensibilidad en una sola métrica, especialmente útil cuando se necesita comparar dos clasificadores.

En diagnóstico médico aunque el dataset esté balanceado, el costo de los errores (como falsos negativos) puede seguir siendo alto. Por eso, en medicina se suele priorizar recall (sensibilidad) o F1-score, no por el balance del dataset, sino por la gravedad de los errores.

Interesa maximizar recall de un algoritmo si el objetivo es identificar todos los casos positivos relevantes. Prioriza el valor F1-score cuando los datos presentan un desequilibrio significativo entre clases positivas y negativas, como ocurre en diagnósticos médicos, donde los falsos negativos no son deseables.

{1: Sí diabetes; 0: No diabetes}

- TN (True Negative)
- TP (True Positive)
- FP (False Positive)

- FN (False Negative)

El total de cada fila muestra todos los positivos pronosticados (TP + FP) y todos los negativos pronosticados (FN + TN), independientemente de la validez. En cambio, el total de cada columna muestra todos los verdaderos positivos (TP + FN) y todos los verdaderos negativos (FP + TN), independientemente de la clasificación del modelo. Ver Cuadro 3.7. (Google Developers, 2025)

Métricas derivadas. A partir de esta matriz podemos calcular:

- Accuracy = $(TP + TN) / \text{Total}$
- Precision = $TP / (TP + FP)$
- Recall = $TP / (TP + FN)$
- F1-score = $2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$

Accuracy (exactitud)

Cuando un tipo de error (FN o FP) es más costoso que el otro - como en diagnóstico médico, donde un falso negativo (FN) puede significar no detectar una enfermedad- que es nuestro caso, no basta con optimizar la accuracy, porque esta métrica trata todos los errores por igual. Es mejor optimizar para una de las otras métricas (F1-score, Precision, Recall). Recall (sensibilidad): mide cuántos casos positivos reales se detectan. Útil cuando los falsos negativos son más peligrosos, como en la detección de diabetes, así que optimizar el recall es lo más prudente. Esto puede implicar usar un umbral más bajo para que el modelo sea más sensible, aunque aumenten los falsos positivos.

Recall (sensibilidad)

Mide cuántos casos positivos reales se detectan. Útil cuando los falsos negativos son más peligrosos, como en la detección de diabetes, así que optimizar el recall es lo más prudente. Esto puede implicar usar un umbral más bajo para que el modelo sea más sensible, aunque aumenten los falsos positivos.

Curva ROC AUC

La curva ROC representa la tasa de falsos positivos frente a la tasa de verdaderos positivos para todos los umbrales posibles: un buen clasificador se mantiene lo más alejado posible de la línea diagonal (la curva del clasificador se sitúa hacia la esquina superior izquierda). Una forma de comparar clasificadores es medir el área bajo la curva (AUC). Un clasificador perfecto tendrá un AUC ROC igual a 1, mientras que un clasificador puramente aleatorio tendrá un AUC ROC igual a 0,5. Scikit-Learn proporciona una función para calcular el AUC ROC (Géron, 2023).

La curva ROC es una representación visual del rendimiento del modelo en todos los umbrales. La AUC permite comparar el rendimiento de dos modelos diferentes, siempre que el conjunto de datos esté aproximadamente equilibrado (nuestro caso). El modelo con mayor área debajo de la curva es el mejor. (Google Developers. (s.f.))

2.4.10. SHAP

Indica la aportación de cada característica al resultado de la predicción del modelo. SHAP (SHapley Additive exPlanations) es un método de interpretación de modelos de aprendizaje automático que permite evaluar cómo

cada variable influye en las predicciones, incluso en modelos complejos o poco transparentes. Calcula la contribución marginal de cada característica (feature) para entender su efecto positivo o negativo en el resultado. En contextos como el diagnóstico médico, esta explicabilidad es muy importante para generar confianza en profesionales y pacientes. (Peng, 2024) A cada característica se le asigna un valor de importancia que representa su contribución al resultado del modelo. Con los valores SHAP y conseguimos dar interpretabilidad a los modelos de machine learning. Tener modelos precisos es necesario, pero también es importante saber interpretarlos y que sean transparentes. Ser capaz de explicar por qué un modelo hizo una predicción concreta ayuda a tratar posibles sesgos, identificar problemas con los datos y justificar las decisiones del modelo. (Awan, 2024)

La falta de interpretabilidad de los modelos de ML para predecir en enfermedades crónicas como la diabetes ha limitado su adopción por parte del mundo sanitario. Para corregirlo se ha creado SHAP.

Con SHAP se dibuja un diagrama donde se muestra la influencia de cada factor en la predicción del modelo. Cita: «Al cuantificar la contribución de cada característica, SHAP permite a los médicos, profesionales de la salud pública o usuarios finales rastrear el razonamiento detrás de una clasificación de alto o bajo riesgo.» (Allani, 2025)

2.4.11. Relación entre RFE y SHAP

La selección de variables y la interpretabilidad del modelo constituyen dos pilares fundamentales en el desarrollo de sistemas predictivos aplica-

dos al ámbito de la salud. En este trabajo, ambos aspectos se abordan de manera complementaria mediante el uso de Recursive Feature Elimination (RFE) y SHapley Additive exPlanations (SHAP), dos técnicas que, aunque conceptualmente distintas, se integran de forma natural dentro del pipeline metodológico.

En primer lugar, RFE se emplea como mecanismo de reducción de dimensionalidad orientado al rendimiento. Su objetivo es identificar el subconjunto de variables que maximiza la capacidad predictiva del modelo. Para ello, RFE entrena iterativamente un estimador y elimina en cada paso las características menos relevantes según la importancia asignada por dicho estimador. Este proceso permite obtener un conjunto compacto de variables que contribuyen de manera significativa a la predicción, reduciendo el ruido, mejorando la estabilidad del modelo y facilitando su posterior interpretación.

Una vez seleccionado este subconjunto óptimo de características, se entrena el modelo final y se aplica SHAP con el fin de analizar la contribución individual de cada variable a las predicciones. Mientras que RFE responde a la pregunta “¿qué variables son necesarias para obtener un buen rendimiento?”, SHAP responde a “¿cómo y en qué dirección influyen estas variables en la predicción?”. De este modo, SHAP proporciona una explicación detallada y cuantificable del impacto de cada característica, tanto a nivel global (importancia media en el conjunto de datos) como a nivel local (contribución en cada individuo).

La combinación de ambas técnicas ofrece una visión completa y coherente del modelo:

- RFE garantiza que el conjunto de variables sean relevantes para construir el modelo.
- SHAP aporta transparencia, interpretabilidad y justificación clínica de los resultados.

Este enfoque integrado resulta especialmente adecuado en contextos sanitarios, donde la interpretabilidad no es un complemento opcional, sino un requisito esencial para la toma de decisiones informadas y responsables.

2.5. Análisis de riesgos

Desbalance de clases. El BRFSS incluye variables de salud con distribuciones desiguales, lo que puede inducir a los modelos a favorecer la clase mayoritaria y generar métricas engañosas (alta precisión global pero baja sensibilidad en la clase minoritaria). Aunque la diabetes tipo 2 está bien representada, la variable objetivo “¿Tiene diabetes?”, presenta un cierto desbalance: predominan los registros de personas sin diabetes frente a los casos positivos. Este sesgo puede reducir el rendimiento en métricas críticas como recall y F1-score, afectando la capacidad de identificar correctamente a individuos con diabetes, algo prioritario en salud pública (Chawla, 2002). Para mitigar el desbalance se aplicó over-sampling en el notebook de la clase minoritaria con SMOTENC y se utilizaron métricas robustas. Se evitó el uso exclusivo de Accuracy y se priorizaron métricas como F1-score, ROC-AUC y curvas de precisión-recall, que reflejan mejor el rendimiento en la clase minoritaria.

Hay que evitar el sobreajuste (overfitting) que ha aparecido debido al feature target leakage (o fuga de información) al entrenar los modelos. Genera métricas engañosamente altas (como precisión perfecta, Accuracy: 1.0), que no se da en datos reales y el modelo no generaliza. (Ver apartado 2.4)

Limitaciones del BRFSS. El BRFSS es una encuesta poblacional basada en datos auto informados, lo que introduce sesgos de memoria, percepción e información. Además, algunas variables presentan categorías poco representadas o distribuciones muy desiguales. Estas limitaciones restringen la precisión absoluta del modelo y su capacidad de reflejar la realidad clínica. Para mitigar este riesgo se aplicaron estrategias de preprocesamiento (imputación de valores faltantes, codificación robusta de variables categóricas), una selección atenta de variables y una interpretación cuidadosa de los resultados, de manera que el modelo refleje patrones poblacionales y no diagnósticos individuales.

Problemas de generalización. Un modelo entrenado en datos del BRFSS puede no generalizar bien a otras poblaciones con características socioeconómicas, culturales o sanitarias diferentes. Esto limita su aplicabilidad directa fuera del ámbito geográfico de la encuesta. Para mitigar este riesgo se recomienda realizar evaluaciones externas en otros conjuntos de datos o datos clínicos, utilizar métricas robustas como ROC-AUC y curvas de precisión-recall para evaluar la capacidad de discriminación, y considerar su interpretación, considerando las predicciones de los modelos como una manera de examinar factores de riesgo más que como un predictor definitivo.

Capítulo 3

Resultados

3.1. Procesamiento de datos

3.1.1. Mitigación de sesgos con AIF360

La función `DISPARATE_IMPACT()` de AIF360 calcula la división entre resultado favorable del grupo no privilegiado y del grupo privilegiado. En el Cuadro 3.3 se muestra como se interpreta `disparate impact`.

- **Antes: `disparate impact` = 1.1028** Indica una ligera ventaja para las mujeres relativa a los hombres. Es un sesgo muy pequeño y cercano a la paridad.
- **Después: `disparate impact` = 1.0** Indica que, tras la corrección con Reweighing, las diferencias entre mujeres y hombres quedaron igual. Hay paridad. (Cuadro 3.1)

Comprobar si hay discriminación por edad:

Antes de la mitigación (para jóvenes y mayores):

Cuadro 3.1: Sesgo según el género

<code>privileged_groups=[{'SEX': 1}] Hombre</code>
<code>unprivileged_groups=[{'SEX': 0}] Mujer</code>
Disparate impact (antes): 1.1028660084117217
Disparate impact (después): 1.0000000000000002

Cuadro 3.2: Sesgo según edad

<code>df['age_group'].map({'jóvenes': 0, 'mayores': 1})</code>
Disparate impact (antes): 0.37542245244029515
Disparate impact (después): 0.9999999999999998

- **Disparate Impact** de 0.375 significa que: el grupo no privilegiado (jóvenes) recibe predicciones positivas (p.ej., “diabetes = 1”) solo el 37.5 % de las veces que las recibe el grupo privilegiado (mayores). Esto indica una fuerte desventaja para los jóvenes.

Después de la mitigación:

- **Disparate Impact** de 0.9999999999999998 que en la práctica es 1.0. Significa que la probabilidad de recibir una predicción positiva es idéntica entre jóvenes y mayores, esto es, el modelo ya no favorece a ningún grupo. El preprocesamiento con Reweighting ha equilibrado completamente la distribución de predicciones entre ambos grupos. (Cuadro 3.2)

El Disparate Impact es el ratio de resultados positivos entre grupos. El Disparate Impact ideal es el que está cercano a 1. Fórmula:

$$\text{Disparate Impact} = \frac{P(\text{resultado positivo} \mid \text{grupo protegido})}{P(\text{resultado positivo} \mid \text{grupo privilegiado})}$$

Cuadro 3.3: Interpretación métrica Disparate Impact de AIF360

Valor	Interpretación
cercano a 1	equidad
< 0.8	posible discriminación contra el grupo no privilegiado
> 1.25	posible discriminación contra el grupo privilegiado.

Con AIF360, al detectar `BINARYLABELDATASETMETRIC` con sesgo por `AGE_GROUP_BIN` edad(usando como atributo protegido), podemos aplicar algoritmos de mitigación de sesgo que generan un nuevo conjunto de datos transformado. Ese dataset se puede exportar a CSV.

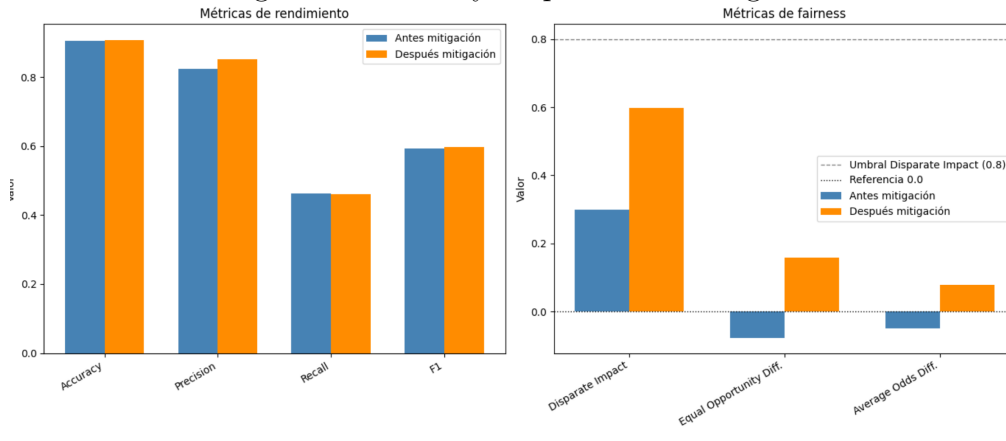
Flujo: **Baseline** → **Mitigación con Reweighing** → **Post-mitigación**

El clasificador usado en las métricas ha sido `LogisticRegression`.

La mitigación con **Reweighing** (ajuste de pesos) ha mejorado mucho la equidad del modelo, aunque el Disparate Impact (0.6) todavía está por debajo del umbral de 0.8, por lo que hay margen de mejora. En la Figura 3.1. se combinan **fairness** y **rendimiento** en una misma tabla comparativa, antes y después de la mitigación con Reweighing.

El Cuadro 3.4 y la Figura 3.2 muestran que la mitigación con AIF360 mejora la equidad de forma clara y mejora ligeramente el rendimiento predictivo. El modelo se vuelve más justo sin sacrificar calidad. La barra, cuanto más cerca esté del umbral 0.8 más justo es. La mitigación no perjudica al modelo; incluso lo mejora ligeramente. El modelo pasa de mostrar fuerte sesgo contra jóvenes a un comportamiento mucho más equilibrado.

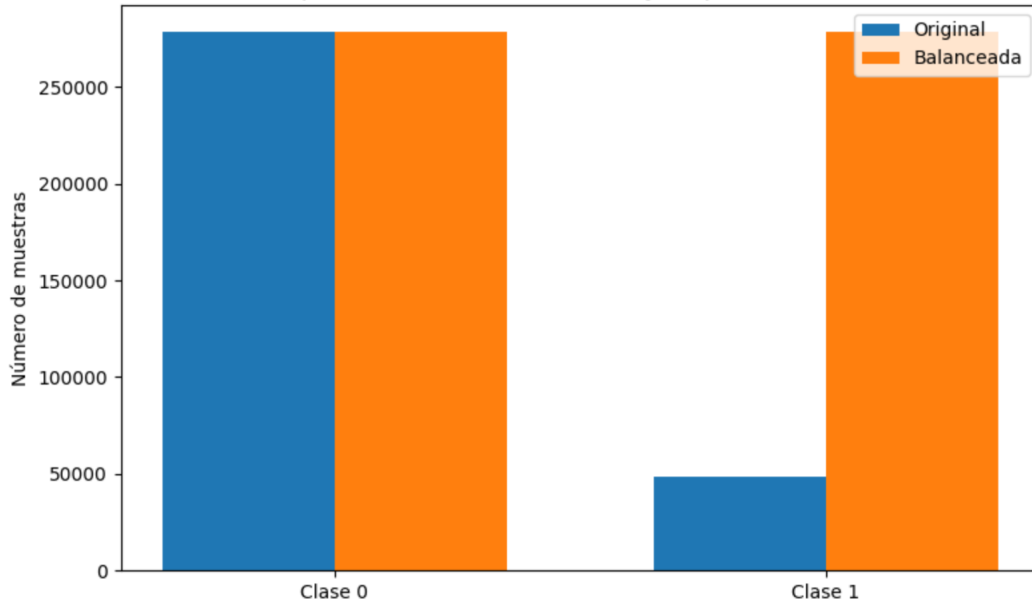
Figura 3.1: Antes y después de la mitigación



Cuadro 3.4: Métricas antes y después de mitigar sesgo por edad

Métrica	Antes mitigación	Después mitigación	Interpretación
Accuracy	0.91	0.91	Ligera mejora
Precision	0.82	0.85	Mejora
Recall (Sensibilidad)	0.46	0.46	Sin cambios
F1-Score	0.59	0.6	Balance entre precisión y recall prácticamente igual.
Disparate Impact	0.3	0.6	Mucho más justo, pero aún lejos del ideal
Equal Opportunity Difference	-0.078	0.16	Pasa de perjudicar a jóvenes a favorecerlos
Average Odds Difference	-0.05	0.08	Sesgo corregido, ligera sobrecompensación

Figura 3.2: Balanceo mediante oversampling con SMOTENC
Comparación de distribución antes y después de SMOTENC



Cuadro 3.5: Las veinte variables más explicativas según RFE

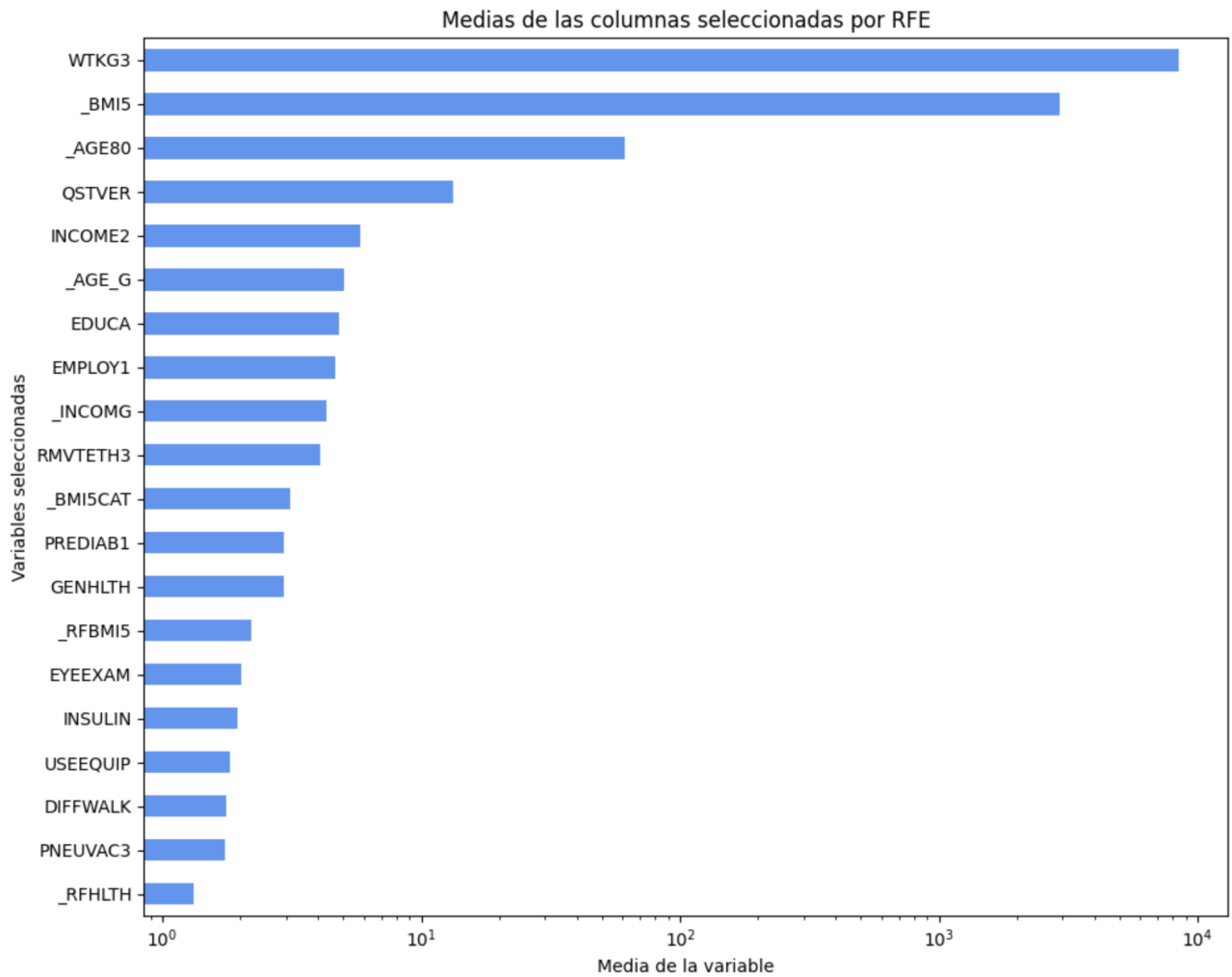
["WTKG3", "_BMI5", "_AGE80", "QSTVER", "INCOME2", "_AGE_G", "EDUCA", "EMPLOY1",
"_INCOMG", "RMVTETH3", "_BMI5CAT", "PREDIAB1", "GENHLTH", "_RFBMI5", "EYEEXAM",
"INSULIN", "USEEQUIP", "DIFFWALK", "PNEUVAC3", "_RFHLTH"]

3.1.2. Reducción de variables

Se escogen las variables más predictivas. Las variables más explicativas calculadas usando RFE se muestran en Figura 3.3, Cuadro 3.5 y 3.6.

Interpretación: El cuadro 3.6 reúne variables que pueden agruparse en varios grupos que ayudan a entender factores de riesgo, comorbilidades, acceso a servicios, hábitos de salud y consecuencias clínicas asociadas a la diabetes tipo 2.

Figura 3.3: Gráfica con las variables más explicativas y su grado de importancia



Cuadro 3.6: Las veinte variables más explicativas según RFE

Variable	Significado
WTKG3	Peso del encuestado en kilogramos.
_BMI5	Índice de Masa Corporal multiplicado por 100.
_AGE80	Edad truncada a 80 años (≥ 80 se codifica como 80).
QSTVER	Versión del cuestionario BRFSS utilizada.
INCOME2	Ingreso anual del hogar en categorías detalladas.
_AGE_G	Edad agrupada en rangos amplios.
EDUCA	Nivel educativo alcanzado.
EMPLOY1	Situación laboral del encuestado.
_INCOMG	Ingreso del hogar agrupado en 5 categorías.
RMVTETH3	Número de dientes permanentes extraídos.
_BMI5CAT	Categoría del IMC (bajo peso, normal, sobrepeso, obesidad).
PREDIAB1	Diagnóstico de prediabetes por un profesional sanitario.
GENHLTH	Autoevaluación de salud general.
_RFBMI5	Indicador de riesgo por IMC (≥ 25 vs < 25).
EYEEXAM	Tiempo desde el último examen ocular.
INSULIN	Uso de insulina en personas con diabetes.
USEEQUIP	Necesidad de equipamiento especial por salud.
DIFFWALK	Dificultad para caminar o subir escaleras.
PNEUVAC3	Vacunación antineumocócica recibida.
_RFHLTH	Indicador de mala salud general (regular/mala).

Cuadro 3.7: Variables más influyentes según SHAP

Variable	Explicación SHAP (interpretación en modelos)
EYEEXAM	Indica el tiempo desde el último examen ocular. Valores que indican <i>más tiempo sin revisión</i> suelen asociarse a peor control de salud, lo que puede aumentar la contribución SHAP hacia riesgo o mala salud.
PREDIAB1	Señala si un profesional ha diagnosticado prediabetes. Un valor positivo suele aumentar la contribución SHAP hacia riesgo metabólico o diabetes futura.
INSULIN	Indica si la persona con diabetes usa insulina. El uso de insulina suele reflejar enfermedad más avanzada, por lo que los valores SHAP tienden a ser positivos en modelos que predicen riesgo o complicaciones.
_AGE_G	Edad agrupada en rangos. Grupos de mayor edad suelen aportar valores SHAP positivos en modelos de riesgo, ya que la edad es un predictor fuerte de enfermedad.
_BMI5CAT	Categoría del IMC (bajo peso, normal, sobrepeso, obesidad). Las categorías de sobrepeso/obesidad suelen generar valores SHAP positivos en modelos de riesgo cardiometabólico.

3.1.3. Balancear la variable objetivo

Una manera de balancear la variable objetivo es mediante técnicas de oversampling con SMOTENC (Figura 3.2). Se crean valores sintéticos de la clase minoritaria (diabetes sí; clase 1). Las dos clases quedan equilibradas en el número de casos en el nuevo conjunto de datos.

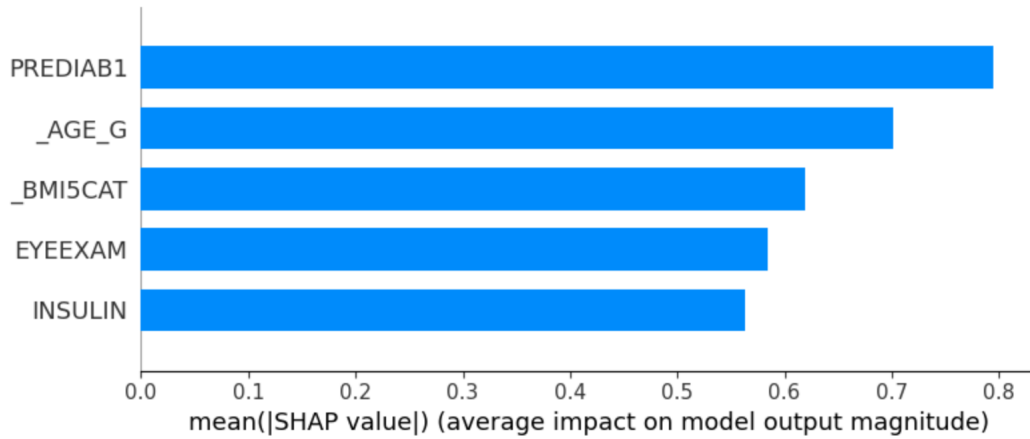
3.1.4. Explicabilidad con SHAP

Variables de entrada y qué predicen

A continuación se muestra una tabla (Cuadro 3.7) con las variables calculadas por SHAP en el modelo CatBoost.

El Cuadro 3.7 muestra que el modelo CatBoost capta variables clínicas, factores como el acceso a servicios sanitarios y la auto percepción de salud. Se ha logrado identificar un conjunto de características informativas, robustas y

Figura 3.4: Diagrama SHAP con XGBoost



socialmente relevantes que sirven tanto para la prevención como para diseñar políticas públicas.

En el diagrama SHAP (Figura 3.4) cada barra horizontal indica cuánto contribuye una variable, en promedio, a las predicciones del modelo. El eje X muestra la magnitud media de los valores SHAP (sin signo), que refleja la influencia global de cada variable. Las variables están ordenadas de mayor a menor importancia. Valores SHAP altos: la variable tiene un fuerte impacto en la predicción (positiva o negativa). Valores SHAP bajos: la variable tiene poco efecto en la decisión del modelo.

Cuadro 3.8: Etapas de mi TFG

<p>Mi flujograma tiene un orden diferente al reflejado en las publicaciones academicas pero efectivo y solvente.</p> <p>Etapas de mi TFG: Transformación de los datos(split train/test;imputar; codificar variables categoricas;normalizado) → Mitigación de sesgos con AIF360 → Seleccionar las variables más predictivas: RFE → Balanceo con SMOTENC → Entrenar modelos → Evaluar el desempeño de los modelos → SHAP.</p> <p>Modelos creados: Regresión Logística, Random Forest, XGBoost, CatBoost</p>

Ashisha, 2024

<p>Pasos que han llevado en este artículo:</p> <p>Adquisición de datos (BRFSS) → Preprocesado (imputación valores faltantes)→ Balancear datos (Random oversampling)→ Extracción de características (Feature Extraction con Boruta y PCA) → Creación de modelos → Evaluación de cada modelo (Accuracy , Precision, Recall, F1 score).</p> <p>Modelos creados: Decision Tree, Random Forest, GradienBoost, Light GBM.</p>

3.2. Estado del arte

En los recuadros siguientes se muestran los pasos y técnicas aplicados al conjunto de datos BRFSS según la literatura científica más reciente. Comparando estas etapas con las seguidas en este Trabajo Fin de Grado (Cuadro 3.9) podemos comprobar que están alineadas.

Cita: «Los conjuntos de datos presentan un alto grado de desequilibrio, con un marcado sesgo hacia una clase en particular. Por lo tanto, el equilibrio de datos y el tratamiento de los datos faltantes son pasos cruciales. [...] valores faltantes [...] datos mediante imputación por la media.» (Ashisha, 2024)

Cita: «Los algoritmos más recientes, como XGBoost y CatBoost, se han posiciona-

Amokun, 2025

Pasos que han llevado en este artículo:

Adquisición de datos (BRFSS) → Preprocesado (imputación valores faltantes) → Balancear datos (SMOTE) → ~~Extracción de características (Feature Extraction con Boruta y PCA)~~ → Creación de modelos → Evaluación de cada modelo (Accuracy, Precision, Recall, F1 score).

Modelos creados: Logistic Regression, Random Forest, Decision Trees, Support Vector Machines, K-Nearest Neighbors, Gradient Boosting Methods, CatBoost, Ensemble Methods

do como líderes en la predicción de enfermedades cardíacas gracias a su capacidad para manejar interacciones complejas y de gran volumen de características, imposibles en algoritmos simples. [...] El desequilibrio de clases [...] tiene como principal efecto sesgar las predicciones hacia la clase mayoritaria, lo que provoca que no se detecten los pacientes en riesgo. [...] Con la aplicación de SMOTE, los modelos mejoraron notablemente su capacidad para predecir enfermedades cardíacas con una precisión mucho mayor. Modelos como Random Forest, CatBoost y XGBoost obtuvieron los mejores resultados, con una precisión, exhaustividad y puntuaciones F1 generales muy altas tanto para la clase mayoritaria como para la minoritaria». (Amokun, 2025)

Cita: «Los algoritmos de aprendizaje automático y las técnicas de aumento de datos pueden desempeñar un papel importante en la identificación temprana de personas con riesgo de diabetes arrojando luz sobre la intrincada red de causas subyacentes. Cada conjunto de datos tiene distintos hiperparámetros que han de ser ajustados para aumentar la calidad de los modelos clasificadores. Es decir, las métricas de los modelos de este TFG no son directamente comparables con los resultados descritos en la literatura científica.» (Chowdhury, 2024)

Pasos que han llevado en este artículo:

Adquisición de datos (BRFSS) → Preprocesado (selección manual de 20 variables) → Balancear datos (SMOTE, ENN, SMOTE-NN, SMOTEN-TOMEK) → Codificar valores nominales (one-hot encoding) → Creación de modelos → Evaluación de cada modelo (Accuracy , Precision, Recall, F1 score) → Ajuste de hiperparámetros.

Modelos creados: Logistic Regression, Gradient Boosting, AdaBoost, and Random Forest.

Chowdhury, 2024

Capítulo 4

Conclusiones y trabajos futuros

Un principio rector en medicina es que en los diagnósticos médicos interesa identificar todos los casos positivos relevantes donde los falsos negativos no son deseables. Es mejor tener un exceso de falsos positivos que serían evaluados posteriormente para confirmarlo.

Al modelo entrenado CatBoost, se le he aplicado SHAP, método para explicar las predicciones del mismo. Como conclusión, el riesgo de diabetes depende tanto de parámetros clínicos como de hábitos de vida, acceso a la atención médica y factores demográficos.

La metodología está alineada con el estado del arte referente al análisis y procesamiento de los datos del BRFSS.

En un principio tenía previsto usar la GPU para entrenar los modelos creyendo que el crearlos sería costoso en tiempo de ejecución pero he comprobado que se crean en segundos o pocos minutos con la CPU. Por lo tanto he usado sólo la CPU multinúcleo para entrenar. Crear el código es más sencillo.

En el conjunto de datos originales había sesgo por edad y sorprendentemente no por género. Se han mitigado el sesgo por edad y actualizado el dataset en consecuencia. La clase no privilegiada eran los jóvenes.

Aparición no prevista de *feature data leakage* en la creación de los primeros modelos. Una vez corregida he podido entrenar los modelos con métricas realistas (no con Accuracy del 1.0).

Como líneas de trabajo futuro que no han podido explorarse en este trabajo y han quedado pendientes: Afinar los hiperparámetros en varios modelos para comprobar cual algoritmo tiene mejor rendimiento respecto CatBoost. Hay que tener en cuenta que para diferentes conjuntos de datos los algoritmos con mejor rendimiento cambian de caso en caso.

Capítulo 5

Glosario

AUC: Área bajo la curva

BRFSS: Behavioral Risk Factor Surveillance System (Sistema de Vigilancia de Factores de Riesgo Conductuales)

CDC: Los Centers for Disease Control and Prevention.

CSV: Comma Separated Values

DAFO: Debilidades, Amenazas, Fortalezas y Oportunidades.

DM2: Diabetes Mellitus Tipo 2.

GPU: unidad de procesamiento gráfico (del inglés graphics processing unit).

GUI: Interfaz gráfica de usuario.

RFE: Recursive Feature Elimination de Scikit-learn.

ROC: Es una curva cuya forma indica la capacidad de un modelo de clasificación binaria para separar las clases positivas de las negativas.

SAS: Statistical Analysis System. Software especializado en análisis estadístico, gestión de datos y minería de datos.

SHAP: SHapley Additive exPlanations. Indica la aportación de cada característica al resultado de la predicción del modelo

SMOTE: Synthetic Minority Over-sampling Technique.

SMOTENC: Synthetic Minority Oversampling Technique for Nominal and Continuous

T2DM: Type 2 Diabetes Mellitus.

Capítulo 6

Bibliografía

Bibliografía

- [1] Abhishek, K., & Abdelaziz, M. (2023). Machine learning for imbalanced data. Packt Publishing. Recuperado de <https://learning.oreilly.com/...>
- [2] AI Fairness 360 Development Team. (n.d.). AI Fairness 360 documentation (aif360 0.6.1). Read the Docs. <https://aif360.readthedocs.io/en/stable/index.html>
- [3] Allani, U. (2025). Interactive diabetes risk prediction using explainable machine learning. arXiv:2505.05683. <https://arxiv.org/pdf/2505.05683>
- [4] Akil, A. A., Alsulaiman, M., & Alghamdi, A. A. (2021). Diagnosis and treatment of type 1 diabetes. Journal of Translational Medicine, 19(1), 1-15. <https://pubmed.ncbi.nlm.nih.gov/33794915/>
- [5] Amokun, R., Arowolo, T., & Eke, J. (2025). Comparative analysis of ML algorithms for heart disease prediction. EasyChair Preprint No. 15706. <https://easychair.org/...>
- [6] Ashisha, G. R., et al. (2024). Random oversampling-based diabetes classification. International Journal of Computational Intelligence Systems, 17(270). doi:10.1007/s44196-024-00678-3

- [7] Awan, A. A. (2024). Introduccion a los valores SHAP. DataCamp.
<https://www.datacamp.com/...>
- [8] Bellamy, R. K. E., et al. (2018). AI fairness 360. arXiv:1810.01943.
[doi:10.48550/arXiv.1810.01943](https://doi.org/10.48550/arXiv.1810.01943)
- [9] Bentejac, C., Csorgo, A., & Martinez-Munoz, G. (2019). A Comparative Analysis of XGBoost. <https://arxiv.org/abs/1911.01914>
- [10] CDC. (2014). Behavioral risk factor surveillance system survey questionnaire.
- [11] Chawla, N. V., et al. (2002). SMOTE. Journal of Artificial Intelligence Research, 16, 321-357. [doi:10.1613/jair.953](https://doi.org/10.1613/jair.953)
- [12] Chen, F., et al. (2025). Type 2 diabetes and asthma incidence. BMC Public Health, 25, 166. [doi:10.1186/s12889-024-21266-2](https://doi.org/10.1186/s12889-024-21266-2)
- [13] Chowdhury, M. M., et al. (2024). ML algorithms and augmentation for diabetes diagnosis. Healthcare Analytics, 5, 100297. [doi:10.1016/j.health.2023.100297](https://doi.org/10.1016/j.health.2023.100297)
- [14] GeeksforGeeks. (2023). Wrapper Methods - Feature Selection. <https://www.geeksforgeeks.org/...>
- [15] Geron, A. (2023). Hands-on machine learning with Scikit-Learn and PyTorch. <https://learning.oreilly.com/...>
- [16] Google Developers. (2025). Umbrales y matriz de confusion. <https://developers.google.com/...>

- [17] Google Developers. (s.f.). ROC and AUC. <https://developers.google.com/...>
- [18] Hancock, J. T., & Khoshgoftaar, T. M. (2020). CatBoost for Big Data. <https://www.researchsquare.com/...>
- [19] Knowler, W. C., et al. (2002). Reduction in the incidence of type 2 diabetes. NEJM, 346(6), 393-403. <https://www.nejm.org/...>
- [20] Lemaitre, G., et al. (2017). Imbalanced-learn. Journal of Machine Learning Research, 18(17), 1-5. <https://jmlr.org/...>
- [21] Martinez-Gonzalez, M. A., et al. (2014). Bioestadística amigable. Elsevier España.
- [22] Mitigation Strategies. International Journal for Research Publication and Seminar, 15, 36-49. <https://www.researchgate.net/...>
- [23] Moccia, C., et al. (2024). Machine learning in causal inference for epidemiology. European Journal of Epidemiology, 39(1), 1-14. [doi:10.1007/s10654-024-01173-x](https://doi.org/10.1007/s10654-024-01173-x)
- [24] Mukherjee, M., & Khushi, M. (2021). SMOTE-ENC. Applied System Innovation, 4(1), 18. [doi:10.3390/asi4010018](https://doi.org/10.3390/asi4010018)
- [25] Li, W., Peng, Y., & Peng, K. (2024). Diabetes prediction model based on GA-XGBoost. PLoS ONE, 19(9), e0311222. [doi:10.1371/journal.pone.0311222](https://doi.org/10.1371/journal.pone.0311222)
- [26] Ramirez Jimenez, O. (2022). Python a fondo. Marcombo.

- [27] Saito, T., & Rehmsmeier, M. (2015). Precision-recall plot vs ROC. PLOS ONE, 10(3), e0118432. [doi:10.1371/journal.pone.0118432](https://doi.org/10.1371/journal.pone.0118432)
- [28] Scikit-learn. (s.f.). Feature selection. <https://scikit-learn.org/...>
- [29] Scikit-learn. (s.f.). SelectFromModel. <https://scikit-learn.org/...>
- [30] UOC Labs. (n.d.). Supervised learning classification. <https://gitlab.uoclabs.uoc.es/...>

Capítulo 7

Anexos

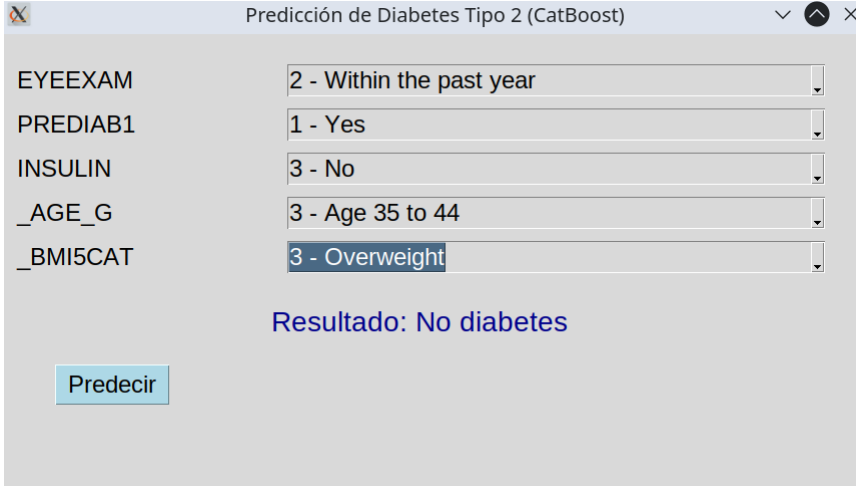
Conversión del conjunto de datos LLCP2014.XPT

El conjunto de datos LLCP2014.XPT se encuentra originalmente en formato XPT (SAS Transport Format), utilizado por la aplicación estadística SAS para el intercambio estructurado de datos entre plataformas. Este formato, aunque normalizado, no es directamente compatible con entornos de análisis en Python, por lo que fue necesario realizar una conversión previa. Para ello, se desarrolló un guion en Python que permite transformar el archivo .XPT al formato CSV, facilitando su posterior tratamiento en bibliotecas como pandas, scikit-learn. Este proceso garantiza la integridad de los datos y permite su integración en flujos de trabajo reproducibles y automatizados.

Aplicación de predicción con interfaz gráfica

He creado una aplicación con Python y Tkinter (Figura 7.1) como ejemplo de uso para cribado (modelo entrenado con CatBoost) para predecir si a partir de unas cinco respuestas si el paciente presenta riesgo de diabetes tipo 2.

Figura 7.1: Aplicación de cribado con interfaz gráfica



Predicción de Diabetes Tipo 2 (CatBoost)

EYEEEXAM	2 - Within the past year
PREDIAB1	1 - Yes
INSULIN	3 - No
_AGE_G	3 - Age 35 to 44
_BMI5CAT	3 - Overweight

Resultado: No diabetes

Predecir

Tkinter es un framework para crear aplicaciones con interfaz gráfica que usa el toolkit Tcl/Tk. Ambos, framework y toolkit, están incluidos por defecto en Python. (Ramírez, 2022)

Instrucciones de instalación de la aplicación

La aplicación Python está creada en un PC1 con sistema operativo Linux/Ubuntu 22.04. La versión de Python de este PC1 es 3.9.18. Para que este programa funcione en otra versión de sistema operativo lo conveniente es crear en el PC2 un entorno virtual Python nuevo. Hay que instalar la versión de Python 3.9.18. Para ello hay que descargar previamente la versión de Python 3.9.18 y compilarla (en el PC2).

Una vez instalada la nueva versión de Python, con el fichero requirements.txt que adjunto se instalan las librerías necesarias. Con pip install se descargan dentro del entorno virtual, sin afectar al sistema global del PC2.

PC1:

```
source app_diabetes_env/bin/activate  
(app_diabetes_env) alex@ryzen5:~/AppGUI_0.9$ pip freeze  
>requirements.txt
```

PC2:

```
python3 -m venv app_diabetes_env  
source app_diabetes_env/bin/activate  
pip install -r requirements.txt
```

- El sistema operativo del PC2 podrá seguir usando su versión de Python aunque no sea compatible con la AppGUI_0.14.py.
- El entorno virtual empleará Python 3.9.18 y tendrá las mismas librerías que en PC1.
- Así se puede trabajar con ambos sin conflictos.