GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Aplicación del Aprendizaje Federado para predecir el nivel de estrés y ansiedad de una minoría desfavorecida en Córdoba (Proyecto E - DUCASS)

Autor: Ángel Cuadrado Serrano

Director: Pablo Pérez Martínez

Co-Directora: Esther Porras Pérez

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

Aplicación del Aprendizaje Federado para predecir el nivel de estrés y ansiedad de una minoría desfavorecida en Córdoba (Proyecto E - DUCASS)

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2024/25 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido

tomada de otros documentos está debidamente referenciada.

Fdo.: Ángel Cuadrado Serrano Fecha:3/7/2025

CUADRADO SERRANO ANGEL - 76750710D

Firmado digitalmente por CUADRADO SERRANO ANGEL - 76750710D
Fecha: 2025.07.03 21:30:28 +02'00'

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Pablo Pérez Martínez Fecha: 3/7/2025

PEREZ MARTINEZ PABLO - 30801943K

Firmado digitalmente por PEREZ MARTINEZ PABLO - 30801943K
DN: C=ES, SERIALNUMBER=IDCES-30801943K, G=PABLO, SN=PEREZ MARTINEZ, CN=PEREZ MARTINEZ PABLO - 30801943K
Razón: Soy el autor de este documento
Ubicación:
Fecha: 2025.07.03 21:17:13+02'00'
Foxit PDF Reader Versión: 2024.1.0

LA CO-DIRECTORA DEL PROYECTO

Fdo.: Esther Porras Pérez Fecha: 3/7/2025

Firmado por PORRAS PEREZ ESTHER - ***4107** el día 03/07/2025 con un certificado emitido por AC FNMT Usuarios

GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Aplicación del Aprendizaje Federado para predecir el nivel de estrés y ansiedad de una minoría desfavorecida en Córdoba (Proyecto E - DUCASS)

Autor: Ángel Cuadrado Serrano

Director: Pablo Pérez Martínez

Co-Director: Esther Porras Pérez

Madrid

Agradecimientos

Antes de comenzar la memoria de este trabajo de fin de grado me gustaría agradecer a D. Pablo Pérez Martínez y a Dña. Esther Porras Pérez por su confianza y apoyo durante todo el proyecto.

APLICACIÓN DEL APRENDIZAJE FEDERADO PARA PREDECIR EL NIVEL DE ESTRÉS Y ANSIEDAD DE UNA MINORÍA DESFAVORECIDA EN CÓRDOBA (PROYECTO E - DUCASS)

Autor: Cuadrado Serrano, Ángel.
Director: Pérez Martínez, Pablo.
Co-Directora: Porras Pérez, Esther.
Entidad Colaboradora: IMIBIC.

RESUMEN DEL PROYECTO

El presente trabajo aborda el desafío de predecir los niveles de ansiedad y depresión en una minoría desfavorecida, utilizando datos sensibles del proyecto E-DUCASS. Se ha desarrollado una metodología de aprendizaje automático que compara un amplio espectro de modelos de regresión y clasificación, desde bases lineales hasta ensamblajes complejos. Los resultados demuestran una alta capacidad predictiva, la cual depende de forma crítica de la comorbilidad entre ansiedad y depresión. Adicionalmente, se ha simulado con éxito un sistema de Aprendizaje Federado, demostrando que es una alternativa viable que logra un rendimiento competitivo protegiendo al mismo tiempo la privacidad de los datos.

Palabras clave: Aprendizaje Federado, Salud Mental, Ansiedad, Depresión, Modelos Predictivos, Privacidad de Datos.

1. Introducción

La salud mental se ha convertido en uno de los mayores desafíos para los sistemas de salud pública a nivel global. Los trastornos de ansiedad y depresión, en particular, presentan una alta prevalencia, afectando de manera desproporcionada a poblaciones en situación de vulnerabilidad social. La detección temprana y la monitorización de estos trastornos son fundamentales para una intervención efectiva, pero a menudo se ven obstaculizadas por la estigmatización y las barreras de acceso. En este contexto, el análisis de datos mediante técnicas de aprendizaje automático ofrece una oportunidad sin precedentes para desarrollar herramientas de cribado y apoyo a la decisión clínica.

Sin embargo, el uso de datos de salud está estrictamente regulado por normativas de privacidad como el Reglamento General de Protección de Datos (RGPD) de la Unión Europea. La necesidad de centralizar la información para entrenar modelos predictivos entra en conflicto directo con el imperativo de proteger la confidencialidad del paciente. Este proyecto se sitúa en la intersección de esta problemática, proponiendo una doble aproximación: primero, desarrollar los modelos predictivos más precisos posibles en un entorno centralizado ideal; y segundo, explorar el Aprendizaje Federado (FL) como un paradigma técnico y ético que permite el entrenamiento colaborativo de modelos sin compartir los datos sensibles. El objetivo final es validar una metodología que sea a la vez potente en su capacidad predictiva y robusta en su garantía de privacidad.

2. Definición del proyecto

El objetivo principal de este trabajo es desarrollar y evaluar un sistema basado en aprendizaje automático capaz de predecir con precisión los niveles de ansiedad y depresión, medidos a través de las puntuaciones de los cuestionarios GAD-7 y PHQ-9, respectivamente. Este objetivo se desglosa en dos perspectivas de modelado:

- Regresión: Predecir la puntuación numérica exacta para cuantificar la magnitud de los síntomas.
- Clasificación: Predecir la categoría clínica de severidad (ej. leve, moderada, severa) para alinear los resultados con la práctica diagnóstica.

El objetivo secundario, de carácter innovador, es simular, implementar y evaluar un sistema de Aprendizaje Federado para la misma tarea predictiva. Esto permite cuantificar el posible "coste" en rendimiento que supone la descentralización de los datos y determinar si el enfoque federado es una alternativa viable para una futura implementación en un entorno real con múltiples centros o fuentes de datos.

El estudio se basa en los datos anonimizados proporcionados por el IMIBIC en el marco del proyecto E-DUCASS, que incluyen un conjunto de variables sociodemográficas, clínicas y de comportamiento de una minoría desfavorecida de Córdoba. La naturaleza sensible de estos datos es la principal justificación para la exploración del paradigma de Aprendizaje Federado.

3. Descripción del modelo/sistema/herramienta

La metodología del proyecto se estructuró en dos fases principales: el modelado centralizado y la simulación federada.

En la fase centralizada, se aplicó un pipeline de preprocesamiento de datos que incluía la imputación de valores nulos, la codificación de variables categóricas y el escalado de características numéricas. Posteriormente, se entrenó y evaluó un amplio espectro de algoritmos, incluyendo:

- Modelos Base Lineales: Regresión Ridge y Regresión Logística, como puntos de referencia.
- Modelos de Ensamblaje: Random Forest, como modelo no lineal robusto.
- Modelos de Gradient Boosting: LightGBM, XGBoost y CatBoost, representando el estado del arte en rendimiento para datos tabulares.
- Meta-Ensamblaje: Un modelo Stacking que combina las predicciones de los modelos anteriores para maximizar la precisión.

En la fase federada, se simuló un entorno con múltiples clientes, cada uno poseedor de un subconjunto de los datos. El sistema se basa en una arquitectura cliente-servidor donde se implementó el algoritmo de Federated Averaging (FedAvg). El proceso iterativo (Ilustración 1) consiste en que el servidor distribuye un modelo global a los clientes, estos lo reentrenan con sus datos locales, y devuelven únicamente las actualizaciones de los parámetros (coeficientes), nunca los datos brutos. El servidor agrega estas actualizaciones para refinar

el modelo global. Para esta simulación se utilizó la Regresión Logística debido a la simplicidad de agregar sus parámetros lineales.

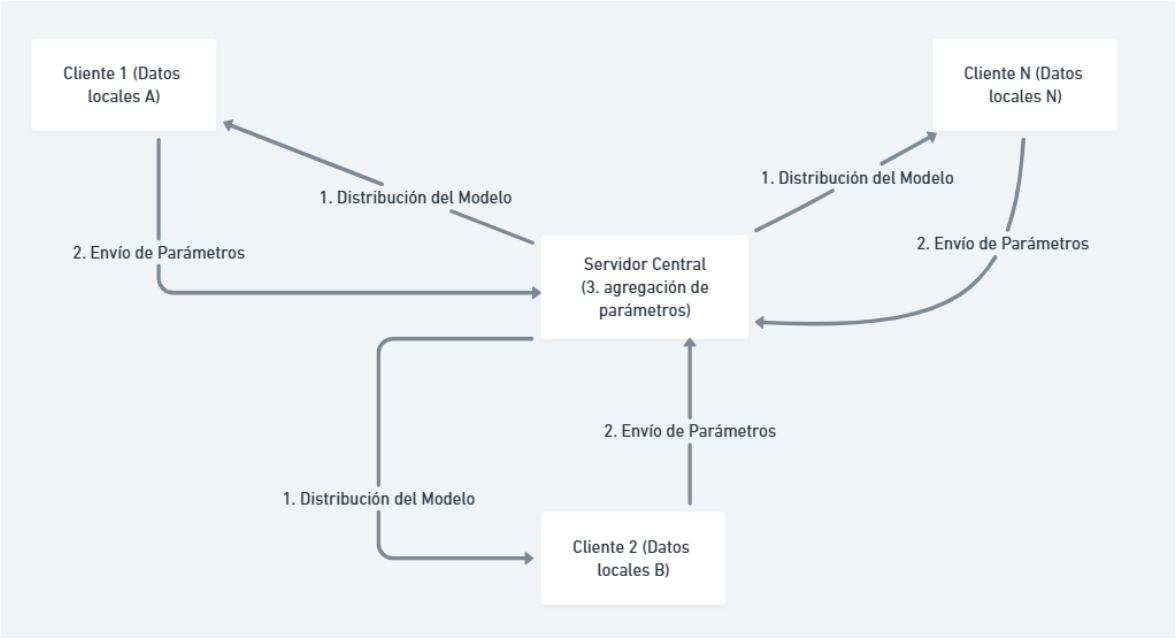


Ilustración 1 Esquema de la arquitectura de Aprendizaje Federado implementada.

4. Resultados

La evaluación exhaustiva de los modelos arrojó varios resultados clave. En el entorno centralizado, los modelos de gradient boosting (especialmente LightGBM y XGBoost) y el ensamblaje Stacking lograron el mayor rendimiento predictivo tanto en las tareas de regresión como de clasificación, superando notablemente a los modelos base.

El hallazgo más significativo fue la validación empírica de la fuerte comorbilidad entre ansiedad y depresión. Se realizaron experimentos eliminando las variables del GAD-7 al predecir el PHQ-9, y viceversa. En ambos casos, el rendimiento de todos los modelos se desplomó a niveles casi aleatorios, demostrando que la presencia de una condición es, con diferencia, el predictor más potente de la otra. Esto sugiere que, con las variables disponibles, no se lograron identificar otros predictores secundarios de fuerza comparable.

En cuanto a la simulación de Aprendizaje Federado, los resultados fueron muy prometedores. Como se observa en la Ilustración 2, el modelo federado logró un rendimiento predictivo competitivo, con una ligera pero aceptable degradación en comparación con su contraparte centralizada en el caso del GAD7 (ansiedad) pero incluso superando a su competidor directo, la Regresión de Ridge en la predicción del PHQ9 (depresión). Este resultado demuestra que es posible construir un modelo colaborativo y preciso sin comprometer la privacidad del individuo, validando el enfoque como una solución práctica y ética para aplicaciones futuras.

Comparativa de Rendimiento Final para PHQ_9_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.6602	2.6131	3.3475
Random Forest	0.6293	2.7237	3.4964
Stacking	0.6155	2.7305	3.5609
XGBoost	0.6113	2.6624	3.5802
CatBoost	0.5970	2.8141	3.6451
Ridge Federado	0.4691	3.5604	4.1840
Ridge	0.3387	3.6274	4.6698

Tabla 1 Comparativa de Rendimiento Final PHQ9

Comparativa de Rendimiento Final para GAD_7_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.5850	2.6441	3.4523
Random Forest	0.5751	2.5419	3.4931
Stacking	0.5681	2.5764	3.5220
CatBoost	0.5617	2.6390	3.5478
XGBoost	0.4615	2.8784	3.9327
Ridge	0.4149	3.1910	4.0992
Ridge Federado	0.4129	3.1196	4.1061

Tabla 2 Comparativa de Rendimiento Final GAD7

5. Conclusiones

Este proyecto ha demostrado con éxito que es factible desarrollar modelos de aprendizaje automático de alta precisión para la predicción de ansiedad y depresión en poblaciones vulnerables. Se ha concluido que el rendimiento predictivo está intrínsecamente ligado a la fuerte comorbilidad entre ambas condiciones, un hallazgo que valida empíricamente el conocimiento clínico a través de métodos computacionales.

La principal contribución de este trabajo es la demostración de que el Aprendizaje Federado no es solo una posibilidad teórica, sino una alternativa práctica y eficaz. El modelo federado simulado alcanzó un rendimiento robusto con una pérdida de precisión mínima, confirmando que se puede lograr un equilibrio favorable entre la utilidad del modelo y la protección de la privacidad de los datos. Este enfoque abre la puerta al desarrollo de herramientas de salud mental colaborativas, éticas y escalables.

Como trabajo futuro, se propone la validación de estos modelos en cohortes más amplias, la exploración de arquitecturas federadas más complejas (ej. con redes neuronales) y la ingeniería de nuevas características para identificar predictores secundarios que puedan enriquecer el modelo más allá de la relación de comorbilidad.

6. Referencias

[1] Khalil, SS, Tawfik, NS y Spruit, M. Exploración del potencial del aprendizaje federado en la investigación sobre salud mental: una revisión sistemática de la literatura. Appl Intell 54 , 1619–1636 (2024). https://doi.org/10.1007/s10489-023-05095-1

[2] Li, J., Jiang, M., Qin, Y. et al. Intelligent depression detection with asynchronous federated optimization. Complex Intell. Syst. 9, 115–131 (2023). https://doi.org/10.1007/s40747-022-00729-2

[3] Rieke, N., Hancox, J., Li, W. et al. The future of digital health with federated learning. npj Digit. Med. 3, 119 (2020). https://doi.org/10.1038/s41746-020-00323-1

APLICACIÓN DEL APRENDIZAJE FEDERADO PARA PREDECIR EL NIVEL DE ESTRÉS Y ANSIEDAD DE UNA MINORÍA DESFAVORECIDA EN CÓRDOBA (PROYECTO E - DUCASS)

Author: Cuadrado Serrano, Ángel.
Supervisor: Pérez Martínez, Pablo.
Co-Supervisor: Porras Pérez, Esther.
Collaborating Entity: IMIBIC.

ABSTRACT

This work addresses the challenge of predicting anxiety and depression levels in a disadvantaged minority using sensitive data from the E-DUCASS project. A machine learning methodology has been developed that compares a broad spectrum of regression and classification models, from linear baselines to complex ensembles. The results demonstrate a high predictive capacity, which is critically dependent on the comorbidity between anxiety and depression. Additionally, a Federated Learning system has been successfully simulated, proving it to be a viable alternative that achieves competitive performance while protecting data privacy.

Keywords: Federated Learning, Mental Health, Anxiety, Depression, Predictive Models, Data Privacy.

1. Introduction

Mental health has become one of the greatest challenges for public health systems globally. Anxiety and depression disorders, in particular, show a high prevalence, disproportionately affecting socially vulnerable populations. The early detection and monitoring of these disorders are fundamental for effective intervention but are often hindered by stigmatization and barriers to access. In this context, data analysis through machine learning techniques offers an unprecedented opportunity to develop screening tools and clinical decision support systems.

However, the use of health data is strictly regulated by privacy regulations such as the General Data Protection Regulation (GDPR) of the European Union. The need to centralize information to train predictive models directly conflicts with the imperative to protect patient confidentiality. This project is situated at the intersection of this issue, proposing a dual approach: first, to develop the most accurate predictive models possible in an ideal centralized environment; and second, to explore Federated Learning (FL) as a technical and ethical paradigm that allows for the collaborative training of models without sharing sensitive data. The final objective is to validate a methodology that is both powerful in its predictive capability and robust in its privacy guarantees.

2. Project Definition

The main objective of this work is to develop and evaluate a machine learning-based system capable of accurately predicting anxiety and depression levels, measured through the scores of the GAD-7 and PHQ-9 questionnaires, respectively. This objective is broken down into two modeling perspectives:

- Regression: Predicting the exact numerical score to quantify the magnitude of symptoms.
- Classification: Predicting the clinical severity category (e.g., mild, moderate, severe) to align the results with diagnostic practice.

The secondary, innovative objective is to simulate, implement, and evaluate a Federated Learning system for the same predictive task. This allows for the quantification of the potential "cost" in performance resulting from data decentralization and helps determine if the federated approach is a viable alternative for future implementation in a real-world setting with multiple centers or data sources.

The study is based on anonymized data provided by IMIBIC within the framework of the E-DUCASS project, which includes a set of sociodemographic, clinical, and behavioral variables from a disadvantaged minority in Córdoba. The sensitive nature of this data is the primary justification for exploring the Federated Learning paradigm.

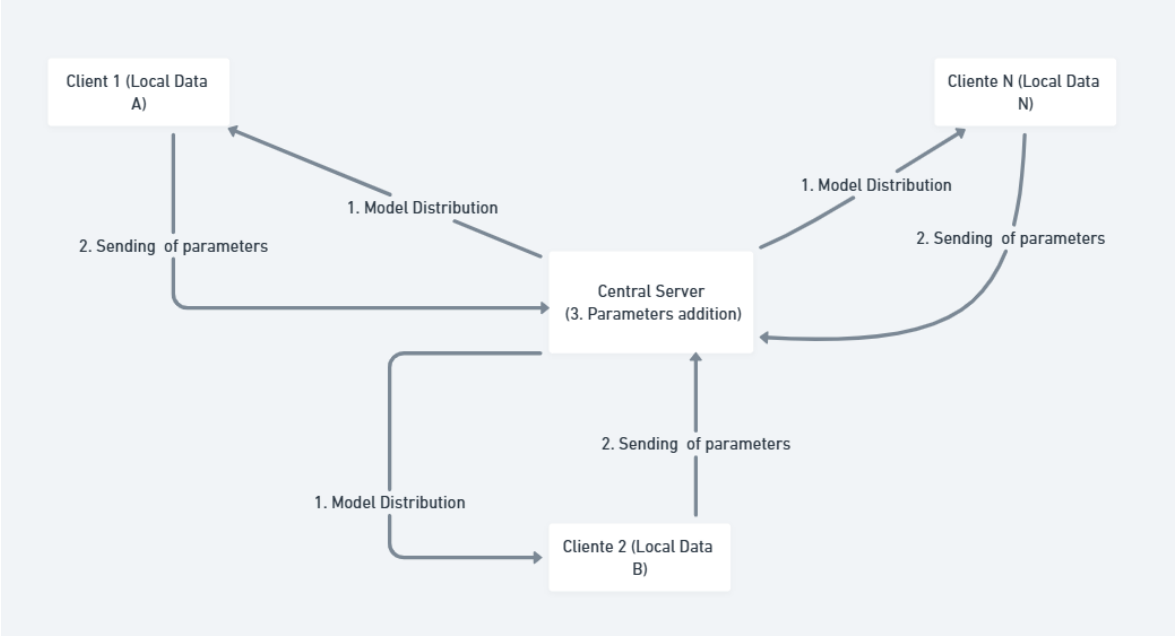
3. Model/System/Tool Description

The project's methodology was structured into two main phases: centralized modeling and the federated simulation.

In the centralized phase, a data preprocessing pipeline was applied, which included the imputation of null values, the encoding of categorical variables, and the scaling of numerical features. Subsequently, a broad spectrum of algorithms was trained and evaluated, including:

- Linear Base Models: Ridge Regression and Logistic Regression, as benchmarks.
- Ensemble Models: Random Forest, as a robust non-linear model.
- Gradient Boosting Models: LightGBM, XGBoost, and CatBoost, representing the state-of-the-art in performance for tabular data.
- Meta-Ensemble: A Stacking model that combines the predictions of the previous models to maximize accuracy.

In the federated phase, an environment with multiple clients was simulated, each holding a subset of the data. The system is based on a client-server architecture where the Federated Averaging (FedAvg) algorithm was implemented. The iterative process (Illustration 1) consists of the server distributing a global model to the clients, who then retrain it on their local data and return only the parameter updates (coefficients), never the raw data. The server aggregates these updates to refine the global model. Logistic Regression was used for this simulation due to the simplicity of aggregating its linear parameters.



Ilustration 2 Schematic of the implemented Federated Learning architecture.

4. Results

The exhaustive evaluation of the models yielded several key results. In the centralized environment, the gradient boosting models (especially LightGBM and XGBoost) and the Stacking ensemble achieved the highest predictive performance in both regression and classification tasks, significantly outperforming the base models.

The most significant finding was the empirical validation of the strong comorbidity between anxiety and depression. Experiments were conducted by removing GAD-7 variables when predicting PHQ-9, and vice versa. In both cases, the performance of all models plummeted to near-random levels, demonstrating that the presence of one condition is, by far, the most potent predictor of the other. This suggests that, with the available variables, no other secondary predictors of comparable strength could be identified.

Regarding the Federated Learning simulation, the results were very promising. As observed in Illustration 2 (in Spanish as is copied from the original content), the federated model achieved competitive predictive performance, with a slight but acceptable degradation compared to its centralized counterpart in the case of GAD-7 (anxiety), but even surpassing its direct competitor, Ridge Regression, in the prediction of PHQ-9 (depression). This result shows that it is possible to build a collaborative and accurate model without compromising individual privacy, validating the approach as a practical and ethical solution for future applications.

Comparativa de Rendimiento Final para PHQ_9_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.6602	2.6131	3.3475
Random Forest	0.6293	2.7237	3.4964
Stacking	0.6155	2.7305	3.5609
XGBoost	0.6113	2.6624	3.5802
CatBoost	0.5970	2.8141	3.6451
Ridge Federado	0.4691	3.5604	4.1840
Ridge	0.3387	3.6274	4.6698

Table 3 Final Performance Comparison for PHQ-9

Comparativa de Rendimiento Final para GAD_7_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.5850	2.6441	3.4523
Random Forest	0.5751	2.5419	3.4931
Stacking	0.5681	2.5764	3.5220
CatBoost	0.5617	2.6390	3.5478
XGBoost	0.4615	2.8784	3.9327
Ridge	0.4149	3.1910	4.0992
Ridge Federado	0.4129	3.1196	4.1061

Table 4 Final Performance Comparison for GAD-7

5. Conclusions

This project has successfully demonstrated that it is feasible to develop high-accuracy machine learning models for the prediction of anxiety and depression in vulnerable populations. It has been concluded that the predictive performance is intrinsically linked to the strong comorbidity between both conditions, a finding that empirically validates clinical knowledge through computational methods.

The main contribution of this work is the demonstration that Federated Learning is not just a theoretical possibility but a practical and effective alternative. The simulated federated model achieved robust performance with minimal accuracy loss, confirming that a favorable balance between model utility and data privacy can be achieved. This approach opens the door to the development of collaborative, ethical, and scalable mental health tools.

Future work proposes the validation of these models on larger cohorts, the exploration of more complex federated architectures (e.g., with neural networks), and the engineering of new features to identify secondary predictors that could enrich the model beyond the comorbidity relationship.

6. References

[1] Khalil, SS, Tawfik, NS y Spruit, M. Exploración del potencial del aprendizaje federado en la investigación sobre salud mental: una revisión sistemática de la literatura. Appl Intell 54 , 1619–1636 (2024). https://doi.org/10.1007/s10489-023-05095-1

[2] Li, J., Jiang, M., Qin, Y. et al. Intelligent depression detection with asynchronous federated optimization. Complex Intell. Syst. 9, 115–131 (2023). https://doi.org/10.1007/s40747-022-00729-2

[3] Rieke, N., Hancox, J., Li, W. et al. The future of digital health with federated learning. npj Digit. Med. 3, 119 (2020). https://doi.org/10.1038/s41746-020-00323-1

Índice de la memoria

Capítulo 1. Introducción 8

Capítulo 2. Descripción de las Tecnologías 10

2.1 SPSS 10

2.2 Excel 10

2.3 Python 11

Capítulo 3. Estado de la Cuestión 13

3.1 Aprendizaje federado en el mundo 13

3.2 Aprendizaje federado en el ámbito de la salud 14

3.3 Aprendizaje federado en el ámbito de la salud mental 15

3.4 Proyecto E - DUCASS 16

Capítulo 4. Definición del Trabajo 17

4.1 Justificación 17

4.1.1 Rigurosidad 17

4.1.2 Privacidad 17

4.1.3 Salud 18

4.2 Objetivos 18

4.2.1 Objetivos Tecnológicos 18

4.2.2 Objetivos Sociales 18

4.3 Metodología 18

4.3.1 Estudio y formación en aprendizaje federado 18

4.3.2 Implementación en código del aprendizaje federado 19

4.3.3 Estudio y limpieza de la base de datos 19

4.3.4 Estudio, prueba y análisis de los diferentes modelos de Machine Learning 19

4.4 Planificación 20

Capítulo 5. Sistema/Modelo Desarrollado 21

5.1 Base de datos 21

5.1.1 Descripción de la base de datos 21

5.1.2 Limpieza de la base de datos 56

5.2 Desarrollo de los modelos de Regresión: Predicción de variables continuas 67
5.2.1 Definiciones 67
5.3 Desarrollo de los modelos de Regresión: Predicción de variables continuas 70
5.3.1 Regresión De Ridge 71
5.3.2 Random Forest Regressor 76
5.3.3 LightGBM Regressor 81
5.3.4 XGBoost Regressor 86
5.3.5 CatBoost Regressor 91
5.3.6 Stacking Regressor 96
5.3.7 Ridge Federado (Aprendizaje Federado Manual) 99
5.4 Desarrollo de los modelos de Clasificación: Predicción de categorías discretas 104
5.4.1 Logistic Regression 105
5.4.2 Random Forest Classifier 110
5.4.3 LightGBM Classifier 115
5.4.4 XGBoost Classifier 120
5.4.5 CatBoost Classifier 125
5.4.6 Stacking Classifier 130
5.4.7 Federated Logistic Regression 133
5.4.8 Comparación General De Los Modelos 136
5.4.9 Resultados Eliminando Las Variables Con Mayor Importancia 137

Capítulo 6. Análisis de Resultados 138
6.1 Precisión por encima del azar en un contexto de pocos datos 138
6.2 Relación estrecha entre ansiedad y depresión 138
6.3 Ausencia de Predictores Secundarios Relevantes 139
6.4 Mejora esperada del modelo con más datos 139
6.5 Utilidad del aprendizaje federado 140
6.6 Validación de hipótesis clínicas desde un enfoque computacional 140
6.7 Resumen del análisis 140

Capítulo 7. Conclusiones y Trabajos Futuros 142
7.1 Conclusiones 142
7.2 Trabajos futuros 143

Capítulo 8. Bibliografía 145

ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS 147

ANEXO II: CÓDIGO UTILIZADO 149

Índice de figuras

Ilustración 1 Esquema de la arquitectura de Aprendizaje Federado implementada. 10
Ilustration 2 Schematic of the implemented Federated Learning architecture. 14
Ilustración 4 Predicciones vs. Valores Reales PHQ9 Modelo Ridge Regression 72
Ilustración 5 Gráfico de Residuos PHQ9 Modelo Ridge Regression 72
Ilustración 6 Importancia Variables PHQ9 Modelo Ridge Regression 73
Ilustración 7 Predicciones vs. Valores Reales GAD7 Modelo Ridge Regression 74
Ilustración 8 Gráfico de Residuos GAD7 Modelo Ridge Regression 74
Ilustración 9 Importancia Variables GAD7 Modelo Ridge Regression 75
Ilustración 10 Predicciones vs. Valores Reales PHQ9 Modelo RF Regressor 77
Ilustración 11 Gráfico de Residuos PHQ9 Modelo RF Regresor 77
Ilustración 12 Importancia Variables PHQ9 RF Regresor 78
Ilustración 13 Predicciones vs. Valores Reales GAD7 Modelo RF Regresor 79
Ilustración 14 Gráfico de Residuos GAD7 Modelo RF Regressor 79
Ilustración 15 Importancia Variables PHQ9 RF Regressor 80
Ilustración 16 Predicciones vs. Valores Reales PHQ9 LightGBM Regressor 82
Ilustración 17 Gráfico de Residuos PHQ9 LightGBM Regressor 82
Ilustración 18 Importancia Variables PHQ9 LightGBM Regressor 83
Ilustración 19 Predicciones vs. Valores Reales GAD7 LightGBM Regressor 84
Ilustración 20 Gráfico de Residuos GAD7 LightGBM Regressor 84
Ilustración 21 Importancia Variables GAD7 LightGBM Regressor 85
Ilustración 22 Predicciones vs. Valores Reales PHQ9 XGBoost Regressor 87
Ilustración 23 Gráfico de Residuos PHQ9 XGBoost Regressor 87
Ilustración 24 Importancia Variables PHQ9 XGBoost Regressor 88
Ilustración 25 Predicciones vs. Valores Reales GAD7 XGBoost Regressor 89
Ilustración 26 Gráfico de Residuos GAD7 XGBoost Regressor 89
Ilustración 27 Importancia Variables GAD7 XGBoost Regressor 90
Ilustración 28 Predicciones vs. Valores Reales PHQ9 CatBoost Regressor 92
Ilustración 29 Gráfico de Residuos PHQ9 CatBoost Regressor 92

Ilustración 30 Importancia Variables PHQ9 CatBoost Regressor 93
Ilustración 31 Predicciones vs. Valores Reales GAD7 CatBoost Regressor 94
Ilustración 32 Gráfico de Residuos GAD7 CatBoost Regressor 94
Ilustración 33 Importancia Variables GAD7 CatBoost Regressor 95
Ilustración 34 Predicciones vs. Valores Reales PHQ9 Stacking Regressor 97
Ilustración 35 Gráfico de Residuos PHQ9 Stacking Regressor 97
Ilustración 36 Predicciones vs. Valores Reales GAD7 Stacking Regressor 98
Ilustración 37 Gráfico de Residuos GAD7 Stacking Regressor 98
Ilustración 38 Predicciones vs. Valores Reales PHQ9 modelo Ridge Federado 102
Ilustración 39 Gráfico de Residuos PHQ9 Modelo Ridge Federado 102
Ilustración 40 Predicciones vs. Valores Reales GAD7 modelo Ridge Federado 103
Ilustración 41 Gráfico de Residuos GAD7 Modelo Ridge Federado 103
Ilustración 42 Matriz de Confusión PHQ9 Logistic Regression 106
Ilustración 43 Importancia de variables PHQ9 Logistic Regression 107
Ilustración 44 Matriz de Confusión GAD7 Logistic Regression 108
Ilustración 45 Importancia de variables GAD7 Logistic Regression 109
Ilustración 46 Matriz de Confusión PHQ9 RF Classifier 111
Ilustración 47 Importancia de variables PHQ9 RF Classifier 112
Ilustración 48 Matriz de Confusión GAD7 RF Classifier 113
Ilustración 49 Importancia de variables GAD7 RF Classifier 114
Ilustración 50 Matriz de Confusión PHQ9 LightGBM Classifier 116
Ilustración 51 Importancia de variables PHQ9 LightGBM Classifier 117
Ilustración 52 Matriz de Confusión GAD7 LightGBM Classifier 118
Ilustración 53 Importancia de variables GAD7 LightGBM Classifier 119
Ilustración 54 Matriz de Confusión PHQ9 XGBoost Classifier 121
Ilustración 55 Importancia de variables PHQ9 XGBoost Classifier 122
Ilustración 56 Matriz de Confusión GAD7 XGBoost Classifier 123
Ilustración 57 Importancia de variables GAD7 XGBoost Classifier 124
Ilustración 58 Matriz de Confusión PHQ9 CatBoost Classifier 126
Ilustración 59 Importancia de variables PHQ9 CatBoost Classifier 127

Ilustración 60 Matriz de Confusión GAD7 CatBoost Classifier 128
Ilustración 61 Importancia de variables GAD7 CatBoost Classifier..... 129
Ilustración 62 Matriz de Confusión PHQ9 Stacking Classifier..... 131
Ilustración 63 Matriz de Confusión GAD7 Stacking Classifier 132
Ilustración 64 Matriz de Confusión PHQ9 Federated Logistic Regressor 134
Ilustración 65 Matriz de Confusión GAD7 Federated Logistic Regressor 135

Índice de tablas

Tabla 1 Comparativa de Rendimiento Final PHQ9 11
Tabla 2 Comparativa de Rendimiento Final GAD7 11
Table 3 Final Performance Comparison for PHQ-9 15
Table 4 Final Performance Comparison for GAD-7 15
Tabla 5 Planificación cronológica del proyecto 20
Tabla 6 Comparativa de Rendimiento Final PHQ9 136
Tabla 7 Comparativa de Rendimiento Final GAD7 136
Tabla 8 Comparativa de Rendimiento Final PHQ9 sin GAD7 137
Tabla 9 Comparativa de Rendimiento Final GAD7 sin PHQ9 137

Capítulo 1. INTRODUCCIÓN

La inteligencia artificial ha venido para revolucionar nuestro planeta, llevando a muchas diferentes industrias al siguiente nivel. La inteligencia artificial está mejorando nuestro mundo, optimizando los procesos internos y externos de las empresas, dando nuevos recursos a la investigación e incluso mejorando nuestro día a día con aplicaciones que nos ayudan a nuestras tareas rutinarias.

No obstante, las inteligencias artificiales no dejan de ser modelos muy bien entrenados que necesitan de una gran cantidad de datos, cuya extracción puede acabar completamente con la privacidad de los usuarios.

Si hay alguna aplicación de la inteligencia artificial que me motiva especialmente es el tema de la salud, especialmente la salud mental. Cada vez son más las personas con diferentes enfermedades mentales. El incremento del déficit de atención y las altas tasas de depresión y estrés son tan solo una pequeña muestra de la preocupante situación de la salud mental de los ciudadanos. Esto concierna a muchos científicos que trabajan cada día para intentar mejorar la vida de aquellos que padecen este tipo de enfermedades, pero en cuanto al cerebro humano aún queda mucho que descubrir.

Personalmente considero que en este campo la inteligencia artificial puede ser de gran ayuda, especialmente a la hora de diagnosticar o prevenir este tipo de enfermedades, y es por ello que quiero empezar a aplicarla en este campo. Sin embargo, como explicaba anteriormente, para realizar buenos modelos de inteligencia artificial necesitamos recopilar muchos datos, lo que puede ser un tema delicado cuando se trata de temas de salud.

Es cierto que nuestra privacidad personal ya es bastante escasa, pues vendemos todos los días nuestros datos con tan solo un par de clics, pero considero que cuando se trata de temas de salud, hay que proteger los datos algo más. Es por ello en este proyecto comprobaremos el funcionamiento del aprendizaje federado para predecir datos de salud, ya que es un tipo

de modelo que, además de otras ventajas, garantiza que los datos del paciente no sean compartidos con nadie.

El aprendizaje federado, en pocas palabras, trata de realizar un modelo de Machine Learning que se entrena en distintos dispositivos, enviando a la máquina central tan solo los resultados de los entrenamientos. Esto hace que no solo necesitemos menos capacidad computacional en nuestra máquina central sino que también respetaremos los datos de todos y cada uno de los clientes ya que no tendremos acceso a ellos, pues no saldrán en ningún momento de sus dispositivos.

En este proyecto, no solo profundizaremos más en las ventajas y desventajas del aprendizaje federado, sino que también compararemos su rendimiento con otros modelos más comunes en Machine Learning.

Todo esto lo haremos de la mano de IMIBIC (Instituto Maimónides de Investigación Biomédica de Córdoba), dentro del proyecto E - DUCASS [1]. IMIBIC trabaja en muchos proyectos distintos, mas uno de sus principales es E - DUCASS, el cual tiene como objetivo favorecer la educación en salud en los tres barrios más vulnerables de Córdoba. Para ello han estado recopilando datos de una población vulnerable en Córdoba durante los últimos dos años. Entre estos datos se incluyen más de 200 columnas, de donde nuestro trabajo será analizar las más interesantes e intentar predecir el nivel de estrés y ansiedad de las personas según el resto de los parámetros. Será aquí donde entre la aplicación del aprendizaje federado y la comparación con otros modelos.

Actualmente la aplicación del aprendizaje federado es pequeña en comparación con otros modelos y es por ello que intentaremos buscarle en este proyecto un hueco en el ámbito de la salud, donde estoy seguro de que puede ser una herramienta muy importante.

Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

Con el objetivo de facilitar la comprensión de los siguientes capítulos, en este apartado se describen en detalle las principales tecnologías, herramientas y lenguajes de programación empleados a lo largo del proyecto. Cada una de estas tecnologías ha sido seleccionada por su idoneidad en relación con las tareas específicas del análisis de datos, visualización, tratamiento estadístico y desarrollo computacional.

2.1 SPSS

SPSS (Statistical Package for the Social Sciences) es un software ampliamente utilizado en las ciencias sociales, diseñado para realizar análisis estadísticos complejos de manera sencilla. En este proyecto, SPSS se ha empleado principalmente en la fase inicial de exploración y visualización de los datos. Su interfaz gráfica intuitiva permite representar gráficamente distribuciones, realizar tablas de frecuencia, y aplicar análisis descriptivos básicos (media, mediana, moda, desviación estándar, etc.) sin necesidad de escribir código.

Además, SPSS facilita el manejo de datos categóricos, como respuestas de cuestionarios, y permite detectar patrones iniciales, valores atípicos o datos faltantes. Todo ello lo convierte en una herramienta adecuada para una primera aproximación al conjunto de datos antes de proceder a un análisis más avanzado.

2.2 EXCEL

Microsoft Excel es una herramienta de hoja de cálculo ampliamente conocida y utilizada en diversos ámbitos, desde la gestión empresarial hasta el análisis de datos. En el contexto de este proyecto, Excel se ha utilizado tanto para una revisión preliminar del conjunto de datos como para tareas básicas de limpieza y organización.

Excel permite visualizar rápidamente grandes volúmenes de datos en formato tabular, así como aplicar filtros, ordenar columnas, y realizar operaciones básicas (sumas, medias, transformaciones de variables, etc.). También ha sido útil para detectar y corregir errores de entrada de datos, y para realizar un control de calidad previo a la carga de los datos en entornos de análisis más complejos como Python o SPSS. Adicionalmente, se ha utilizado para la exportación e importación de datos entre formatos (.xlsx, .csv), garantizando la interoperabilidad con otras plataformas.

2.3 PYTHON

Python es un lenguaje de programación de alto nivel, ampliamente adoptado en el campo del análisis de datos por su sintaxis clara, versatilidad y la gran cantidad de librerías especializadas. En este proyecto, Python ha sido la herramienta principal para el procesamiento, análisis avanzado, visualización y modelado de datos.

A continuación, se detallan algunas de las librerías y entornos de trabajo más relevantes:

Pandas: Esta biblioteca proporciona estructuras de datos flexibles y potentes (como los DataFrames) que permiten manipular grandes conjuntos de datos de forma eficiente. Ha sido utilizada para la limpieza de datos, gestión de valores nulos, transformación de variables, combinación de datasets y análisis estadísticos básicos.

NumPy: Aunque de uso más general para cálculos matemáticos, NumPy ha servido como soporte para operaciones vectorizadas de alto rendimiento y manipulación de arrays multidimensionales.

Matplotlib y Seaborn: Estas bibliotecas han sido empleadas para la creación de gráficos y visualizaciones estadísticas. Matplotlib permite construir visualizaciones personalizadas, mientras que Seaborn, basada en esta, ofrece gráficos más sofisticados y con mayor facilidad de uso para datos categóricos o correlaciones entre variables.

Scikit-learn: En las fases más avanzadas del proyecto, Scikit-learn ha sido utilizada para la construcción y evaluación de modelos de aprendizaje automático supervisado. Esta biblioteca facilita la implementación de algoritmos como regresión logística, árboles de decisión, k-vecinos más cercanos, entre otros, además de proporcionar herramientas para la validación cruzada, selección de características y análisis de métricas.

Jupyter Notebooks: El entorno interactivo de Jupyter ha sido clave para el desarrollo incremental del análisis, permitiendo combinar texto, visualizaciones y código en un solo documento. Esto ha facilitado tanto la documentación del proceso como la reproducibilidad de los resultados.

XGBoost, LightGBM y CatBoost: Estas tres bibliotecas representan implementaciones de vanguardia del algoritmo de *gradient boosting*. Se han utilizado para construir los modelos predictivos de más alto rendimiento del proyecto. A diferencia de los ensamblajes nativos de Scikit-learn, estas librerías están altamente optimizadas para la velocidad y la eficiencia en el uso de memoria, permitiendo entrenar modelos complejos sobre grandes volúmenes de datos en tiempos reducidos.

SHAP (SHapley Additive exPlanations): Para abordar el problema de la interpretabilidad en los modelos complejos ("caja negra"), se ha empleado la biblioteca SHAP. Esta herramienta, basada en la teoría de juegos, permite cuantificar la contribución exacta de cada una de las variables a una predicción específica. Su uso ha sido fundamental para explicar los resultados de los modelos de *boosting*, aportando transparencia y permitiendo entender qué factores son los que más influyen en las decisiones del modelo a nivel tanto global como individual.

Capítulo 3. ESTADO DE LA CUESTIÓN

3.1 APRENDIZAJE FEDERADO EN EL MUNDO

Actualmente son muchas las empresas que están implementando inteligencia artificial en distintos proyectos, mas muy pocas lo hacen apoyándose en aprendizaje federado. No obstante, la aplicación de este algoritmo es cada vez mayor, especialmente en el campo de la salud como veremos más adelante, pues, como hemos explicado anteriormente, no necesita compartir los datos y respeta por ende la privacidad del paciente. Sin embargo, su aplicación fuera de este campo también es interesante.

Algunas empresas como Google llevan años implementando el aprendizaje federado. En el caso de Google su aplicación más afamada es en GBoard, su teclado para dispositivos móviles y tabletas [2]. Este intenta predecir las palabras que el usuario va a escribir apoyándose en un modelo entrenado con aprendizaje federado. No solo es positivo aquí el uso del aprendizaje federado por la privacidad del usuario, sino que en este caso en especial la capacidad computacional que se ahorra es enorme, pues almacenar todas y cada una de las palabras en orden que escribe el usuario y entrenarlas puede conllevar un gran coste si se hace en una sola máquina central.

Apple, por su parte, hace uso del aprendizaje federado para entrenar a su Inteligencia Artificial Siri, ya que también les ahorra mucha capacidad computacional y ofrece al usuario una gran privacidad a la vez que proporciona un gran rendimiento [3].

Así, podríamos nombrar un número muy alto de empresas que utilizan esta tecnología, y cada vez son más dado a las ventajas ya mencionadas.

Inteligencia artificial en el ámbito de la salud

La inteligencia artificial también ha venido para revolucionar el sector de la salud. Son muchas las empresas y laboratorios que están implementando la inteligencia artificial para

muchos proyectos en este ámbito ya que no solo es una herramienta para optimizar procesos dentro de la investigación, sino que es un recurso complementario a otras tecnologías previamente utilizadas.

Uno buen ejemplo es el de DeepMind, una filial de Google, que ha colaborado con el Servicio Nacional de Salud del Reino Unido para desarrollar algoritmos de IA que analizan datos médicos y ayudan en la detección temprana de enfermedades. Un ejemplo es el uso de IA para la detección de enfermedades oculares [4].

Otro caso de aplicación de la inteligencia artificial en el mundo de la medicina es el de Microsoft Research, que desarrolló Project InnerEye, que aplica técnicas de aprendizaje automático para la segmentación de imágenes médicas, facilitando diagnósticos más precisos y rápidos [5].

Así, son muchos las empresas y laboratorios de investigación que están aplicando de forma exitosa la inteligencia artificial en sus proyectos, pero muchas de ellas aún lo hacen haciendo uso de algoritmos de Machine Learning distintos al aprendizaje federado.

3.2 APRENDIZAJE FEDERADO EN EL ÁMBITO DE LA SALUD

EL aprendizaje federado se está aplicando cada vez más en el ámbito de la salud, donde parece que puede ser de gran interés sobre todo al no necesitar compartir los datos de los pacientes y poder respetar así su privacidad al completo. Son muchos los proyectos donde ya se está aplicando, ejemplo de ello es el caso de NVIDIA. Esta empresa hace uso de esta tecnología para el entrenamiento de un modelo de reconocimiento de imágenes médicas [6]. Aunque otras empresas como Owkin están aplicando esta tecnología para otros modelos de salud y biotecnología [7]. Por ejemplo, Owkin formó parte del consorcio MELLODDY (Machine Learning Ledger Orchestration for Drug Discovery), que incluía a 10 compañías farmacéuticas y otros socios. Este proyecto aplicó el aprendizaje federado para entrenar modelos de inteligencia artificial en conjuntos de datos sin compartir información propietaria, con el objetivo de mejorar el descubrimiento de fármacos [8].

Además, no tan solo grandes empresas extranjeras están aplicando el aprendizaje federado en el sector de la salud. En España ya son varios proyectos donde se está utilizando este algoritmo. Un ejemplo claro es el de los hospitales 12 de Octubre y Ramón y Cajal de Madrid, y el Hospital de Sant Pau de Barcelona, que colaboran para crear un buen modelo de análisis de imágenes radiológicas a la vez que respetan la privacidad de los pacientes gracias al aprendizaje federado [9].

3.3 APRENDIZAJE FEDERADO EN EL ÁMBITO DE LA SALUD MENTAL

En cuanto a la aplicación del aprendizaje federado en enfermedades mentales cada vez son más los estudios referentes a la aplicación de esta tecnología. No obstante, pocos son muy concluyentes y tan solo unos pocos son llevados a la práctica, al menos hasta 2023 [10]. Como podemos ver en el artículo citado la mayoría de estos estudios se centran en la depresión, mientras que ninguno lo hace en los niveles de estrés.

La mayoría de los resultados de los estudios realizados alrededor de las enfermedades mentales y el aprendizaje federado son positivos, aunque muy pocos, como he mencionado antes, se han llevado o se podrían llevar a la realidad. No solo eso, sino que muy pocos estudios tienen una base de datos tan robusta como la que nos proporciona IMIBIC, que toma no solo datos a través de formularios, sino que cuenta con datos médicos y otro tipo de mediciones muy profesionales como es el GAD-7, que explicaremos más adelante.

Uno de los estudios más interesantes que podemos encontrar es el de “Intelligent depression detection with asynchronous federated optimization” [11] donde, con datos extraídos de redes sociales, se aplica el aprendizaje federado para detectar la depresión de los usuarios.

3.4 PROYECTO E - DUCASS

En cuanto al proyecto E - DUCASS [1], que como hemos explicado antes es un proyecto que está llevando acabo el IMIBIC y tiene como objetivo favorecer la educación en salud en los tres barrios más vulnerables de Córdoba, consta de una gran cantidad de información recogida durante más de dos años.

El proyecto E - DUCASS trata de educar principalmente con recursos tecnológicos a esta minoría vulnerable en Córdoba, tratando de mejorar así la salud de estos. El proyecto, como podríamos imaginar, después de dos años ya consta de muchos avances; y está trabajando en una aplicación para utilizarla como plataforma educativa y mejorar la recolecta de datos.

Capítulo 4. DEFINICIÓN DEL TRABAJO

4.1 JUSTIFICACIÓN

A la vista del capítulo anterior podemos observar como cada vez son más los estudios que se hacen sobre la aplicación del aprendizaje federado en el mundo de la salud, aunque muy pocos son concluyentes en el ámbito de la salud mental. No solo eso, sino que no podemos encontrar ningún caso en el que se esté aplicando el aprendizaje federado en la realidad, más allá de la simulación, para detectar niveles de estrés y depresión. Y los estudios que encontramos son en su mayoría poco rigurosos en cuanto a la base de datos.

Es por ello por lo que se realiza este proyecto, por su significado en cuanto a la mejora de la rigurosidad, de la privacidad; y la salud.

4.1.1 RIGUROSIDAD

La base de datos desarrollada por el IMIBIC es una oportunidad única para demostrar de una forma rigurosa el funcionamiento del aprendizaje federado en el ámbito de la salud mental, ya que se ha desarrollado de una forma rigurosa y profesional, a lo largo de dos años, en uno de los centros de investigación médica más prestigiosos de Andalucía.

No solo es este el incentivo del proyecto, pues con esta base de datos se podrían hacer una infinidad de estudios de interés, sino que la motivación del proyecto va más allá.

4.1.2 PRIVACIDAD

Como explicaba al comienzo de la memoria, uno de los pilares más importantes de esta investigación es respetar los datos del paciente. Hoy en día es muy poca la privacidad que tenemos como usuarios, pero en el tema de la salud, especialmente en el tema de la salud mental, considero importante que se respete la privacidad del individuo, y eso tan solo lo podemos conseguir al completo de la mano del aprendizaje federado.

4.1.3 SALUD

Por último, son muchas las personas que padecen de enfermedades mentales en el mundo, y son muchas también las que no se diagnostican. Esto es un problema que me concierna a nivel personal, y es por ello que quiero enfocar este proyecto al ámbito de la salud mental, pues considero que es un campo donde hay mucho por avanzar y donde creo que la inteligencia artificial puede ser una gran protagonista en dicho avance.

4.2 OBJETIVOS

Dentro de este proyecto perseguiremos distintos objetivos, entre los que podemos encontrar:

4.2.1 OBJETIVOS TECNOLÓGICOS

- Explorar, aprender y dominar una base de datos en SPSS.
- Desarrollar distintos algoritmos de predicción con Python.
- Buscar una nueva aplicación para el aprendizaje federado e investigar su potencial.
- Implementar un algoritmo de Machine Learning dentro de una aplicación.

4.2.2 OBJETIVOS SOCIALES

- Detectar los factores que favorecen la depresión y el estrés para su mejor prevención.
- Respetar la privacidad de los datos a la vez que se trabaja en mejorar los modelos de predicción.

4.3 METODOLOGÍA

Este proyecto consta de varias secciones, entre las que se encuentran:

4.3.1 ESTUDIO Y FORMACIÓN EN APRENDIZAJE FEDERADO

Esta etapa consiste en una formación profunda a través del aprendizaje federado. Durante esta habrá que realizar distintos cursos online para una buena formación, mientras se complementa con la lectura de papers científicos y una investigación profunda.

4.3.2 IMPLEMENTACIÓN EN CÓDIGO DEL APRENDIZAJE FEDERADO

Esta etapa está influenciada por la carencia de la base de datos definitiva. Durante ella implementaremos de distintas formas el aprendizaje federado a nivel de código, con la intención de aprender y familiarizarse con las nuevas librerías. Para ello se usarán otras bases de datos de prueba. Esta fase, como la anterior, no se recogen en la memoria pues no forman parte del proyecto final, sino de la formación necesaria para poder completarlo.

4.3.3 ESTUDIO Y LIMPIEZA DE LA BASE DE DATOS

Durante esta fase se analizará a fondo la base de datos. Limpiando posibles filas o columnas incompletas o erróneas que puedan perjudicar al resultado final de los modelos. La base de datos es muy extensa en cuanto a diferentes variables, por lo que el análisis llevará su tiempo.

4.3.4 ESTUDIO, PRUEBA Y ANÁLISIS DE LOS DIFERENTES MODELOS DE MACHINE LEARNING

Durante esta etapa se desarrollarán los distintos modelos de Machine Learning y se compararán con los resultados del aprendizaje federado de forma detallada y específica.

4.4 PLANIFICACIÓN

La planificación temporal del proyecto ha sido la siguiente:

DICIEMBRE	ENERO	FEBRERO	MARZO	ABRIL	MAYO
FORMACIÓN					
	ESTUDIO DEL APRENDIZAJE FEDERADO E IMPLEMENTACIÓN EN CÓDIGO				
		BASE DE DATOS			
			DESARROLLO Y ESTUDIO DE LOS MODELOS		
		REDACCIÓN			

Tabla 5 Planificación cronológica del proyecto

Capítulo 5. SISTEMA/MODELO DESARROLLADO

A lo largo de este capítulo describiremos el desarrollo del proyecto desde el inicio hasta el final. Este capítulo se dividirá en distintas etapas, que representan las distintas partes que realizarán para llegar al resultado final, empezando por el análisis y la limpieza de la base de datos y terminando por el estudio detallado del aprendizaje federado para la predicción de las variables deseadas en el proyecto.

5.1 BASE DE DATOS

En este apartado detallaremos todo lo necesario alrededor de la base de datos, haciendo primero una descripción de esta y terminando por una limpieza y preparación de los datos para entrenar los modelos posteriormente.

5.1.1 DESCRIPCIÓN DE LA BASE DE DATOS

La base de datos con la que trabajaremos nos la proporciona el IMIBIC (Instituto Maimónides de Investigación Biomédica de Córdoba), que ha recolectado datos de una población desfavorecida de Córdoba durante más de dos años con el objetivo de mejorar la educación alimenticia de estas personas, como hemos explicado en el apartado 3.5 cuando describimos el proyecto E – DUCASS. No obstante, nosotros haremos un uso distinto de esta base de datos. Aprovechando la gran cantidad de variables que nos ofrece la tabla, intentaremos predecir el nivel de estrés y depresión de las personas que forman parte del estudio según las demás variables. Además, he de aclarar aquí que aunque en el estudio se recogen datos de los últimos dos años nosotros tan solo usaremos los resultados del primero, pues en el segundo no se recogen los niveles de estrés y ansiedad.

Una vez descrita la procedencia y el contexto de nuestra base de datos vamos a pasar a ver sus principales características y a describir las distintas variables.

5.1.1.1 Características de la base de datos

Para hacer un análisis inicial de la base de datos, se ha desarrollado un script en Python que permite obtener una visión general y completa del conjunto de datos. Este análisis preliminar es fundamental para entender la estructura de la información, detectar posibles problemas de calidad (como valores nulos o duplicados), y orientar las decisiones en el proceso de limpieza y modelado posterior.

El código se ha implementado utilizando las bibliotecas Pandas y NumPy, y tiene como objetivo identificar las siguientes características:

- Número total de filas y columnas del conjunto de datos.
- Nombre de las variables y tipo de datos asociado a cada una.
- Número de valores nulos por variable y su porcentaje sobre el total.
- Presencia de filas duplicadas.
- Estadísticas descriptivas para variables numéricas (media, desviación típica, máximos, mínimos, etc.).
- Estadísticas para variables categóricas (frecuencia, moda, cardinalidad).
- Porcentaje de valores únicos por columna.
- Muestra de los valores más frecuentes en variables categóricas.

A continuación, se muestra el código utilizado para este análisis:

```
import pandas as pd
import numpy as np

def analizar_base_de_datos(df):
    print("="*60)
    print("▶ INFORMACIÓN GENERAL DEL DATAFRAME")
    print("="*60)
    print(f"- Número de filas: {df.shape[0]}")
    print(f"- Número de columnas: {df.shape[1]}")
    print(f"- Columnas: {list(df.columns)}")
    print(f"- Duplicados: {df.duplicated().sum()} filas duplicadas\n")

    print("="*60)
    print("▶ TIPOS DE DATOS POR COLUMNA")
    print("="*60)
    print(df.dtypes.value_counts())
    print(df.dtypes)

    print("\n" + "="*60)
    print("▶ VALORES NULOS")
    print("="*60)
    nulos = df.isnull().sum()
    porcentaje_nulos = (nulos / len(df)) * 100
    nulos_df = pd.DataFrame({'Nulos': nulos, '% Nulos': porcentaje_nulos})
    print(nulos_df[nulos_df['Nulos'] > 0].sort_values(by='Nulos',
ascending=False))

    print("\n" + "="*60)
    print("▶ ESTADÍSTICAS DESCRIPTIVAS (NUMÉRICAS)")
    print("="*60)
    print(df.describe(include=[np.number]).T)

    print("\n" + "="*60)
    print("▶ ESTADÍSTICAS DESCRIPTIVAS (CATEGÓRICAS)")
    print("="*60)
    print(df.describe(include=['object', 'category']).T)

    print("\n" + "="*60)
    print("▶ CARDINALIDAD DE CADA COLUMNA")
    print("="*60)
    cardinalidades = df.nunique().sort_values(ascending=True)
    print(cardinalidades)
```

```
    print("\n" + "="*60)
    print("▶ PORCENTAJE DE VALORES ÚNICOS POR COLUMNA")
    print("="*60)
    print((df.nunique() / df.shape[0] *
100).sort_values(ascending=False).round(2))

    print("\n" + "="*60)
    print("▶ EJEMPLO DE VALORES ÚNICOS EN VARIABLES CATEGÓRICAS")
    print("="*60)
    for col in df.select_dtypes(include=['object', 'category']).columns:
        print(f"{col}: {df[col].unique()[:10]}{'...' if df[col].nunique() >
10 else ''}")

df = pd.read_excel('Base_Angel_28_3_25.xlsx')
analizar_base_de_datos(df)
```

Como resultado de la ejecución del código obtenemos el siguiente resultado:

```
============================================================
▶ INFORMACIÓN GENERAL DEL DATAFRAME
============================================================
- Número de filas: 460
- Número de columnas: 240
- Columnas: ['CODIGO_ID', 'ID_CLUSTER', 'GRUPO_INTERVENCION',
'FECHA_NACIMIENTO', 'FECHA_EDAD_AÑO1', … , 'GAD7_AÑO1_CATEGORIAS',
'GAD7_AÑO1_CAT_MOD_SEV', 'GAD7_AÑO1_CAT_MIN_OTHER']
- Duplicados: 0 filas duplicadas

============================================================
▶ TIPOS DE DATOS POR COLUMNA
============================================================
float64           205
object             20
int64              13
datetime64[ns]      2
Name: count, dtype: int64
CODIGO_ID                          object
ID_CLUSTER                         object
GRUPO_INTERVENCION                  int64
FECHA_NACIMIENTO           datetime64[ns]
FECHA_EDAD_AÑO1            datetime64[ns]
                                     ...
GAD7_AÑO1_7                        float64
GAD7_AÑO1_TOTAL                    float64
```

```
GAD7_AÑO1_CATEGORIAS              float64
GAD7_AÑO1_CAT_MOD_SEV             float64
GAD7_AÑO1_CAT_MIN_OTHER           float64
Length: 240, dtype: object

============================================================
▶ VALORES NULOS
============================================================
                                  Nulos    % Nulos
EMBARAZADA_SI_NO_AÑO1               457  99.347826
EXFUMADOR_AÑO_1                     403  87.608696
OCUPACION_JORNADA_AÑO_1             398  86.521739
OCUPACION_TIPO_AÑO_1                396  86.086957
BEBEDOR_TIPO_AÑO_1                  351  76.304348
...                                 ...        ...
SMET_HIPERTENSION_AÑO1               28   6.086957
EDUCACION_AÑO_1                      27   5.869565
TALLA_AÑO_1                          25   5.434783
ANALÍTICA_HA_DESAYUNADO              18   3.913043
ORIGEN_VALORES_ANALITICOS_AÑO_1      18   3.913043

[205 rows x 2 columns]

============================================================
▶ ESTADÍSTICAS DESCRIPTIVAS (NUMÉRICAS)
============================================================
                          count       mean        std    min   25%   50%
75%  \
GRUPO_INTERVENCION        460.0   1.000000   0.824463   0.00   0.0   1.0
2.0
EDAD_AÑO1                 460.0  33.012688  14.959387  12.00  18.0  33.0
44.0
TABACO_AÑO_1              422.0   0.919431   0.941597   0.00   0.0   1.0
2.0
EXFUMADOR_AÑO_1            57.0   7.099474   8.082739   0.17   1.0   3.0
12.0
FUMADOR_AÑO_1             161.0  12.245342   7.331251   0.50   7.0  10.0
20.0
...                         ...        ...        ...    ...   ...   ...
...
GAD7_AÑO1_7               424.0   0.830189   0.989034   0.00   0.0   1.0
1.0
GAD7_AÑO1_TOTAL           424.0   7.415094   6.006487   0.00   3.0   6.0
12.0
GAD7_AÑO1_CATEGORIAS      423.0   2.063830   1.029513   1.00   1.0   2.0
3.0
GAD7_AÑO1_CAT_MOD_SEV     421.0   0.318290   0.466367   0.00   0.0   0.0
1.0
GAD7_AÑO1_CAT_MIN_OTHER   422.0   0.623223   0.485153   0.00   0.0   1.0
1.0
```

```
                                 max
GRUPO_INTERVENCION               2.0
EDAD_AÑO1                       84.0
TABACO_AÑO_1                     3.0
EXFUMADOR_AÑO_1                 26.0
FUMADOR_AÑO_1                   40.0
...                              ...
GAD7_AÑO1_7                      3.0
GAD7_AÑO1_TOTAL                 48.0
GAD7_AÑO1_CATEGORIAS             4.0
GAD7_AÑO1_CAT_MOD_SEV            1.0
GAD7_AÑO1_CAT_MIN_OTHER          1.0

[218 rows x 8 columns]

============================================================
▶ ESTADÍSTICAS DESCRIPTIVAS (CATEGÓRICAS)
============================================================
                                          count unique       top freq
CODIGO_ID                                   460    460    ED0197    1
ID_CLUSTER                                  460    162   EDF0001    6
OCUPACION_TIPO_AÑO_1                         64     37   Albañil    6
AMIGOS_FAMILIA_ENTRE_SEMANA_HORAS_AÑO_1     460     13    #NULL!  216
AMIGOS_FAMILIA_FIN_DE_SEMANA_HORAS_AÑO_1    460     22    #NULL!  175
DORMIR_L_V_AÑO_1                            460     19  23:00:00  101
DORMIR_S_D_AÑO_1                            460     23  00:00:00  100
DESPERTAR_L_V_AÑO_1                         460     39  07:00:00  106
DESPERTAR_S_D_AÑO_1                         460     26  10:00:00   74
HORAS_DE_SUEÑO_L_V_AÑO_1                    460     44  08:00:00   75
HORAS_DE_SUEÑO_S_D_AÑO_1                    460     31  09:00:00   78
MEDIA_HORAS_DE_SUEÑO_AÑO_1                  460     65  09:00:00   50
SIESTA_TIEMPO_L_V_AÑO_1                     460     15    #NULL!  170
SIESTA_TIEMPO_S_D_AÑO_1                     460     15    #NULL!  209
DESAYUNO_L_V_AÑO_1                          460     36    #NULL!   82
DESAYUNO_S_D_AÑO_1                          460     31    #NULL!   99
ALMUERZO_L_V_AÑO_1                          460     24  15:00:00  125
ALMUERZO_S_D_AÑO_1                          460     23  15:00:00  126
CENA_L_V_AÑO_1                              460     21  22:00:00  122
CENA_S_D_AÑO_1                              460     21  22:00:00  143

============================================================
▶ CARDINALIDAD DE CADA COLUMNA
============================================================
FECHA_EDAD_AÑO1                      1
EMBARAZADA_SI_NO_AÑO1                1
SCORE7_IDEAL_NO_IDEAL_AÑO1           1
CONSUMO_AZUCAR_AÑO_1                 2
CONSUMO_COMIDA_PRECOCINADA_AÑO_1     2
                                   ...
```

```
PESO_AÑO_1                          323
WHtR_AÑO1                           400
IMC_AÑO_1                           418
FECHA_NACIMIENTO                    451
CODIGO_ID                           460
Length: 240, dtype: int64

============================================================
▶ PORCENTAJE DE VALORES ÚNICOS POR COLUMNA
============================================================
CODIGO_ID                       100.00
FECHA_NACIMIENTO                 98.04
IMC_AÑO_1                        90.87
WHtR_AÑO1                        86.96
PESO_AÑO_1                       70.22
                                 ...
ENFERMEDAD_CANCER_AÑO1            0.43
GAD7_AÑO1_CAT_MIN_OTHER           0.43
FECHA_EDAD_AÑO1                   0.22
EMBARAZADA_SI_NO_AÑO1             0.22
SCORE7_IDEAL_NO_IDEAL_AÑO1        0.22
Length: 240, dtype: float64

============================================================
▶ EJEMPLO DE VALORES ÚNICOS EN VARIABLES CATEGÓRICAS
============================================================
CODIGO_ID: ['ED0028' 'ED0483' 'ED0017' 'ED0250' 'ED0296' 'ED0372' 'ED0373'
'ED0375'
 'ED0378' 'ED0453']...
ID_CLUSTER: ['EDF0007' 'EDF0162' 'EDF0004' 'EDF0079' 'EDF0097' 'EDF0124'
'EDF0125'
 'EDF0126' 'EDF0149' 'EDF0152']...
OCUPACION_TIPO_AÑO_1: ['Autonómo' 'Ayuda a domicilio' nan 'Ayuda a'
'Limpiado' 'Comercial'
 'Ayuda domicilio' 'Camarero' 'Jardinero' 'Teleoperadora']...
```

De esta manera podemos hacernos una primera idea de la estructura y sacar las primeras concusiones con respecto a nuestra base de datos:

En primer lugar, se destaca la dimensión del dataset: cuenta con 460 observaciones y 240 variables, lo cual implica una alta dimensionalidad. Esta relación entre número de columnas y registros puede representar un desafío para muchos algoritmos tradicionales de aprendizaje automático, especialmente aquellos sensibles al sobreajuste o que no incorporan mecanismos de regularización. Por ello, será fundamental considerar estrategias de selección

de características o técnicas de reducción de dimensionalidad (como Análisis de Componentes Principales, PCA) antes de aplicar modelos complejos.

En cuanto a los tipos de datos, se observa una clara predominancia de variables numéricas continuas (205 columnas tipo float64), complementadas por variables categóricas (object, 20 columnas), enteros (int64, 13 columnas) y fechas (datetime, 2 columnas). Este perfil es favorable para la aplicación de modelos estadísticos y técnicas de machine learning que requieren entradas numéricas. Sin embargo, se identifican columnas con un porcentaje extremadamente alto de valores nulos, como EMBARAZADA_SI_NO_AÑO1 (99.3%), EXFUMADOR_AÑO_1 (87.6%) u OCUPACION_JORNADA_AÑO_1 (86.5%), cuya utilidad debe evaluarse críticamente. Estas variables podrían descartarse si no aportan información suficiente o imputarse si resultan clínicamente relevantes.

Además, el elevado número de variables sugiere la existencia de columnas potencialmente problemáticas, tales como aquellas con alta granularidad, codificación textual inconsistente (por ejemplo, respuestas categóricas tipo “Sí”, “si”, “SI”) o redundancia estructural (como ocurre con preguntas individuales y totales en cuestionarios tipo PHQ-9). También se han identificado columnas cuya utilidad es meramente administrativa o identificadora (e.g., CODIGO_ID, ID_CLUSTER), las cuales podrían ser excluidas del análisis clínico pero conservadas como claves para fusiones o trazabilidad de los datos.

Finalmente, se confirma que no existen filas duplicadas, lo cual constituye una buena señal en términos de calidad de los datos. Las columnas de tipo fecha pueden ser transformadas para derivar variables relevantes como edad o antigüedad. En resumen, este conjunto de datos ofrece una rica variedad de dimensiones relacionadas con salud, estilo de vida, hábitos y marcadores clínicos, lo que representa un gran potencial para estudios complejos. No obstante, será imprescindible implementar un proceso riguroso de limpieza y depuración, incluyendo la revisión de valores nulos, estandarización de categorías, y una reducción prudente de variables, como paso previo a cualquier modelo predictivo de salud mental o síndrome metabólico.

5.1.1.2 Descripción de las variables

La base de datos incluye las siguientes variables:

CODIGO_ID: CODIGO ID. Es una variable categórica nominal, representada como string.

SEXO: Sexo. Es una variable categórica nominal. Categorías: 0 = "Hombre", 1 = "Mujer".

ID_CLUSTER: Codigo de familia. Es una variable categórica nominal, representada como string.

GRUPO_INTERVENCION: Grupo de intervención. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Anual sin intervención", 1 = "Tradicional Seminarios", 2 = "Digital".

FECHA_NACIMIENTO: Fecha de nacimiento. Es una variable continua, representada como date.

FECHA_EDAD_AÑO1: Fecha toma muestras año 1. Es una variable continua, representada como date.

EDAD_AÑO1: Edad. Es una variable continua, representada como numeric.

TABACO_AÑO_1: Hábito tabaquico año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca fumador", 1 = "Exfumador", 2 = "Fumador activo", 3 = "Fumador otros". En fumador otros se refiere por ejemplo a vapers.

EXFUMADOR_AÑO_1: Años sin fumar. Es una variable continua, representada como numeric. En el caso de que en la variable anterior sea Exfumador, es decir un 1, se rellena esta variable con los años sin formar, si son meses se calcula, valor númerico.

FUMADOR_AÑO_1: Número de cigarrillos al día. Es una variable continua, representada como numeric. En el caso de que en la variable anterior sea Fumador activo, es decir un 2.

ALCOHOL_AÑO_1: Consumo de alcohol año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca bebedor", 1 = "Exbebedor", 2 = "Bebedor".

BEBEDOR_TIPO_AÑO_1: Tipo de bebida alcoholica año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Cerveza", 1 = "Vino", 2 = "Bebidas destiladas", 3 = "Cerveza y Bebidas destiladas", 4 = "Vino y Bebidas destiladas", 5 = "Cerveza y Vino", 6 = "Cerveza, vino y bebidas destiladas". Esta se responde en el caso de que sea Bebedor.

BEBEDOR_FREQ_AÑO_1: Frecuencia de consumo año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Todos los días", 1 = "Fines de semana", 2 = "Ocasionalmente". Esta se responde en el caso de que sea Bebedor.

BEBEDOR_CANTIDAD_AÑO_1: Cuando bebe cuanto vasos alcohol. Es una variable continua, representada como numeric. Esta se responde en el caso de que sea Bebedor.

BEBEDOR_COMIDAS_AÑO_1: Consumo de alcohol en las comidas año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si". Esta se responde en el caso de que sea Bebedor.

EDUCACION_AÑO_1: Último nivel de estudios superados año 1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ninguna", 1 = "Primaria", 2 = "Secundaria o Grado Medio", 3 = "Universitaria".

OCUPACION_AÑO_1: Ocupación año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Desempleado",1 = "Activo",2 = "Jubilado", 3 = "Estudiante", 4 = "Ama de casa u otros".

OCUPACION_TIPO_AÑO_1: Tipo de trabajo. Es una variable categórica nominal, representada como string.

OCUPACION_JORNADA_AÑO_1: Número de horas de trabajo a la semana. Es una variable continua, representada como numeric.

INGRESOS_AÑO_1: Ingresos medios en el domicilio año 1 en euros. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 500", 1 = "500-1000", 2 = "500-1000". Cuantos dinero (euros) se ingresa en el hogar en total al mes.

CONSUMO_AGUA_AÑO_1: Litros de agua al día. Es una variable continua, representada como numeric.

CONSUMO_CAFE_AÑO_1: Tazas de café al día. Es una variable continua, representada como numeric.

CONSUMO_INFUSIONES_AÑO_1: Infusiones al día. Es una variable continua, representada como numeric.

CONSUMO_ACEITE_OLIVA_AÑO_1: Cucharadas de aceite de oliva al día. Es una variable continua, representada como numeric.

CONSUMO_FRUTA_AÑO_1: Piezas de fruta al día. Es una variable continua, representada como numeric.

CONSUMO_VERDURA_AÑO_1: Consumo de verdura al día. Es una variable continua, representada como numeric.

CONSUMO_CEREALES__AÑO_1: Consumo de pan, cereales, arroz o pasta al día. Es una variable continua, representada como numeric.

CONSUMO_CEREALES_INTEGRALES_AÑO_1: Consumo de cereales integrales al día. Es una variable continua, representada como numeric.

CONSUMO_LACTEOS_AÑO_1: Consumo de leche desnatada, queso fresco y yoghurt al dia. Es una variable continua, representada como numeric.

CONSUMO_REFRESCOS_AÑO_1: Consumo de refrescos azucarados/edulcorados a la semana (litros). Es una variable continua, representada como numeric.

CONSUMO_AZUCAR_AÑO_1: Adición de azúcar a bebidas (café, té, colacao) año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

CONSUMO_COMIDA_PRECOCINADA_AÑO_1: Intenta evitar consumir comida precocinada año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

CONSUMO_COMIDA_BAJA_SAL_AÑO_1: Al comer fuera intenta elegir comidas bajas en sal año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

CONSUMO_CANTIDAD_SAL_AÑO_1: Al cocinar en casa evita añadir sal año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

CONSUMO_PICOTEO_AÑO_1: Picotear entre comidas año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

CONSUMO_PATATAS_AÑO_1: Consumo de patatas (guisadas, asadas o fritas) a la semana. Es una variable continua, representada como numeric.

CONSUMO_FRUTOS_SECOS_AÑO_1: Consumo de frutos secos a la semana. Es una variable continua, representada como numeric.

CONSUMO_PESCADO_AÑO_1: Consumo de pescado a la semana. Es una variable continua, representada como numeric.

CONSUMO_CARNE_ROJA_AÑO_1: Consumo de carne roja (ternera, cerdo, cordero) a la semana. Es una variable continua, representada como numeric.

CONSUMO_CARNE_PROCESADA_AÑO_1: Consumo de carne procesada (hamburguesas, salchichas, embutidos) a la semana. Es una variable continua, representada como numeric.

CONSUMO_CARNE_BLANCA_AÑO_1: Consumo de carne blanca (pollo, pavo, conejo) a la semana. Es una variable continua, representada como numeric.

CONSUMO_HUEVOS_AÑO_1: Número de huevos a la semana. Es una variable continua, representada como numeric.

CONSUMO_LEGUMBRES_AÑO_1: Consumo de legumbres a la semana. Es una variable continua, representada como numeric.

CONSUMO_DULCES_AÑO_1: Número de dulces (galletas, pasteles, magdalenas, donuts) a la semana. Es una variable continua, representada como numeric.

CONSUMO_SNACKS_AÑO_1: Número de paquetes de snacks (patatas fritas, gusanitos, palomitas) a la semana. Es una variable continua, representada como numeric.

CONSUMO_SOFRITO_AÑO_1: Consumo de sofrito (cebolla, ajo, tomate, aceite de oliva...) a la semana. Es una variable continua, representada como numeric.

AMIGOS_FAMILIA_ENTRE_SEMANA_AÑO_1: Suele quedar con familia o amigos entre semana año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si".

AMIGOS_FAMILIA_ENTRE_SEMANA_HORAS_AÑO_1: Numero de horas entre semana. Es una variable continua, representada como date. Si responde 1 a la anterior.

AMIGOS_FAMILIA_FIN_DE_SEMANA_AÑO_1: Suele quedar con familia o amigos en fin de semana año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

AMIGOS_FAMILIA_FIN_DE_SEMANA_HORAS_AÑO_1: Numero de horas en fin de semana. Es una variable continua, representada como date. Si responde 1 a la anterior.

TALLA_AÑO_1: Altura en cm. Es una variable continua, representada como numeric.

PESO_AÑO_1: Peso en kg. Es una variable continua, representada como numeric.

IMC_AÑO_1: IMC. Es una variable continua, representada como numeric.

SOBREPESO_OBESIDAD_ARTICULO_OMS_AÑO1: Clasificación sobrepeso y obesidad según cortes OMS Año1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= "No sobrepeso ni obesidad", 1= "Sobrepeso (IMC=25-29,9) y P85-P97", 2= "Obesidad (IMC≥30) y P>97".

SOBREPESO_OBESIDAD_CDC_ARTICULO_AÑO1: Clasificación sobrepeso y obesidad todos según cortes CDC Año1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= "No sobrepeso ni obesidad", 1= "Sobrepeso (IMC=25-29,9) y P85-P95", 2= "Obesidad (IMC≥30) y P>95".

PERIMETRO_CINTURA_AÑO1: nan. Es una variable continua, representada como numeric. Medición, variable en cm.

GRASA_CORPORAL_AÑO_1: nan. Es una variable continua, representada como numeric. Es el porcentaje de grasa corporal.

MUJER_EDAD_GRASA_MUSC_AÑO1: Clasificación edad para cálculo grasa corporal y músculo año 1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 1=Mujeres de 18-39 años, 2= Mujeres de 40-59 años, 3= Mujeres de 60-80 años.

HOMBRE_EDAD_GRASA_MUSC_AÑO1: Clasificación edad para cálculo grasa corporal y músculo año 1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 1=Hombres de 18-39 años, 2= Hombres de 40-59 años, 3= Hombres de 60-80 años.

REF_GRASA_CORPORAL_AÑO1: Valores de referencia Grasa Corporal (%) Año 1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0=Bajo porcentaje de grasa corporal, 1=Porcentaje Normal de grasa corporal, 2=Porcentaje Elevado de grasa corporal, 3=Porcentaje Muy Elevado de grasa corporal

MASA_MUSCULAR_AÑO_1: nan. Es una variable continua, representada como numeric. Es el porcentaje de músculo.

REF_MUSCULO_AÑO1: Valores de referencia Músculo esquelético (%) Año 1. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= Bajo porcentaje de músculo esquelético, 1= Porcentaje Normal de músculo esquelético, 2= Porcentaje Elevado de músculo esquelético, 3= Porcentaje Muy Elevado de músculo esquelético.

MEDIA_PRESION_ARTERIAL_SISTOLICA_AÑO_1: nan. Es una variable continua, representada como numeric.

MEDIA_PRESION_ARTERIAL_DIASTOLICA_AÑO_1: nan. Es una variable continua, representada como numeric.

MEDIA_PULSACIONES_AÑO_1: nan. Es una variable continua, representada como numeric.

GLUCOSA_AÑO_1: nan. Es una variable continua, representada como numeric.

COLESTEROL_TOTAL_AÑO_1: nan. Es una variable continua, representada como numeric.

LDL_COLESTEROL_AÑO_1: nan. Es una variable continua, representada como numeric.

TRIGLICERIDOS_AÑO_1: nan. Es una variable continua, representada como numeric.

NON_HDL_COLESTEROL_AÑO_1: nan. Es una variable continua, representada como numeric.

HDL_COLESTEROL_AÑO_1: nan. Es una variable continua, representada como numeric.

SMET_OBESIDAD_ABDOMINAL_AÑO1: Obesidad abdominal Criterios NCEP-ATPIII. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0=No tiene obesidad abdominal, 1= Tiene obesidad abdominal

SMET_HIPERTENSION_AÑO1: Presion arterial alta Criterios NCEP-ATPIII. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = No tiene hipertensión, 1= Tiene hipertensión o presión arterial alta.

SMET_TRIGLICERIDOS_AÑO1: Altos TAG Criterios NCEP-ATPIII. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No presenta TAG elevados, 1 = Presenta TAG elevados

SMET_HDL_AÑO1: Bajos HDL Criterios NCEP-ATPIII. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = Tiene adecuados niveles HDL, 1= No tiene adecuados niveles HDL

SMET_GLUCOSA_AÑO1: Glucosa elevada Criterios NCEP-ATPIII. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = No presenta glucosa elevada, 1= Presenta glucosa elevada

SMET_CRITERIOS_SUMA_AÑO1: Suma Criterios NCEP-ATPIII. Es una variable continua, representada como numeric.

CLASIF_SMET_ATPIII_AÑO1: Clasificación SMET (cumple los 3 criterios-Si Smet) Año1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 criterios", 1= "3 o más criterios"

SMET_AO_IDF_AÑO1: Obesidad abdominal Criterios IDF. Es una variable categórica nominal, representada como numeric. 0=No tiene obesidad abdominal, 1= Tiene obesidad abdominal

SMET_BP_IDF_AÑO1: Presion arterial alta Criterios IDF. Es una variable categórica nominal, representada como numeric. 0 = No tiene hipertensión, 1= Tiene hipertensión o presión arterial alta.

SMET_TAG_IDF_AÑO1: Altos TAG Criterios IDF. Es una variable categórica nominal, representada como numeric. 0= No presenta TAG elevados, 1 = Presenta TAG elevados

SMET_HDL_IDF_AÑO1: Bajos HDL Criterios IDF. Es una variable categórica nominal, representada como numeric. 0 = Tiene adecuados niveles HDL, 1= No tiene adecuados niveles HDL

SMET_GLUC_IDF_AÑO1: Glucosa elevada Criterios IDF. Es una variable categórica nominal, representada como numeric. 0 = No presenta glucosa elevada, 1= Presenta glucosa elevada

SIND_IDF_AÑO1: Clasificacion por IDF. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No Smet, 1= Si Smet (Obesidad abdominal + 2 o mas del resto de criterios). En esta clasificación para que tenga sindrome metabolico tiene que presenter obesidad abdominal y la suma de 2 o más del resto de criterios.

WHtR_AÑO1: Es solo para niños para detectar obesidad abdominal. Es una variable continua, representada como numeric.

WHrR_Clasif_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= “WHtR < 0.5 (no obesidad abdominal)”, 1= “WHtR > o igual 0.5 (obesidad abdominal)”

INDICE_CT_HDL_AÑO_1: nan. Es una variable continua, representada como numeric.

INDICE_CT_HDL_CLASIFICACION_AÑO_1: Clasificación Indice CT/HDL Año 1. Es una variable categórica nominal, representada como numeric.

ORIGEN_VALORES_ANALITICOS_AÑO_1: Clasificación: datos de perfil lípidco o datos de analítica Año 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: {0, Original (toma de muestras)}....

ANALÍTICA_HA_DESAYUNADO: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: {0, No ha desayunado}....

EMBARAZADA_SI_NO_AÑO1: nan. Es una variable categorica nominal, representada como numeric.

PASEO_TRANQUILO_DIAS_AÑO_1: Días al mes que pasea tranquilo (camino trabajo, paseo en bici). Es una variable continua, representada como numeric.

PASEO_TRANQUILO_MINUTOS_AÑO_1: Minutos de paseo tranquilo al día. Es una variable continua, representada como numeric.

CAMINAR_DEPRISA_DIAS_AÑO_1: Días al mes que camina deprisa. Es una variable continua, representada como numeric.

CAMINAR_DEPRISA_MINUTOS_AÑO_1: Minutos de caminar deprisa al día. Es una variable continua, representada como numeric.

CAMINAR_CAMPO_DIAS_AÑO_1: Días al mes que camina campo a través. Es una variable continua, representada como numeric.

CAMINAR_CAMPO_MINUTOS_AÑO_1: Minutos al día que camina campo a través. Es una variable continua, representada como numeric.

SUBIR_ESCALERAS_DIAS_AÑO_1: Días al mes que sube las escaleras. Es una variable continua, representada como numeric.

SUBIR_ESCALERAS_PISOS_AÑO_1: Número de pisos que sube al día. Es una variable continua, representada como numeric.

HUERTO_JARDÍN_DIAS_AÑO_1: Días al mes que trabaja en huerto o jardín. Es una variable continua, representada como numeric.

HUERTO_JARDÍN_MINUTOS_AÑO_1: Minutos al día que trabaja en huerto o jardín. Es una variable continua, representada como numeric.

EJERCICIO_FISICO_DIAS_AÑO_1: Días a la semana que realiza ejercicio físico. Es una variable continua, representada como numeric.

EJERCICIO_FISICO_MINUTOS_AÑO_1: Minutos al día que realiza ejercicio físico. Es una variable continua, representada como numeric.

TIPO_EJERCICIO_FISICO_AÑO_1: Tipo ejercicio fisico. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Máquinas gimnasio", 1 = "Fútbol", 2 = "Baloncesto", 3 = "Fútbol y Baloncesto", 4 = "Educación física", 5 = "Correr", 7 = "Baile", 8 = "Gimnasio y Baile", 9 = "Natación", 10 = "Natación, baile y máquinas", 11 = "Fútbol y Bicicleta", 13 = "Bicicleta", 14 = "Pádel", 15 = "Tenis", 16 = "Fútbol y máquinas", 17 = "Natación y voleybol", 18 = "Fútbol, máquinas y correr", 19 = "Voleybol", 20 = "Correr y máquinas", 21 = "Balonmano y Baloncesto", 22 = "Boxeo y máquinas",23 = "Educación física y fútbol",24 = "Educación física y Voleibol",26 = "Fútbol y Badminton",27 = "Artes marciales",29 = "Correr y Fútbol",30 = "Torear",31 = "Correr y bicicleta",32 = "Baloncesto y Voleibol",33 = "Bicicleta y Baile",34 = "Rugby y Futbol",35 = "Boxeo",36 = "Taichi y Gimnasio",37 = "Futbol y Voleibol"

DORMIR_L_V_AÑO_1: Hora de dormir de lunes a viernes. Es una variable continua, representada como date.

DORMIR_S_D_AÑO_1: Hora de dormir sábado y domingo. Es una variable continua, representada como date.

DESPERTAR_L_V_AÑO_1: Hora de despertarse de lunes a viernes. Es una variable continua, representada como date.

DESPERTAR_S_D_AÑO_1: Hora de despertarse sábado y domingo. Es una variable continua, representada como date.

HORAS_DE_SUEÑO_L_V_AÑO_1: Horas de sueño entre semana. Es una variable continua, representada como date.

HORAS_DE_SUEÑO_S_D_AÑO_1: Horas de sueño en fin de semana. Es una variable continua, representada como date.

MEDIA_HORAS_DE_SUEÑO_AÑO_1: Media horas de sueño en la semana. Es una variable continua, representada como date.

CALCULO_HORAS_SUEÑO_LV_AÑO1: Horas de sueño entre semana En númerico. Es una variable continua, representada como numeric.

CALCULO_HORAS_SUEÑO_SD_AÑO1: Horas de sueño en fin de semana en número. Es una variable continua, representada como numeric.

CALCULO_MEDIA_HORAS_DE_SUEÑO_AÑO_1: Media horas de sueño en la semana en número. Es una variable continua, representada como numeric.

SIESTA_L_V_AÑO_1: Duerme siesta entre semana. Es una variable categorica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si".

SIESTA_S_D_AÑO_1: Duerme siesta en fin de semana. Es una variable categorica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si".

SIESTA_TIEMPO_L_V_AÑO_1: Horas de siesta entre semana. Es una variable continua, representada como date.

SIESTA_TIEMPO_S_D_AÑO_1: Horas de siesta en fin de semana. Es una variable continua, representada como date.

DIAS_SIESTA_L_D_AÑO_1: Días que duerme siesta en total. Es una variable continua, representada como numeric.

DESAYUNA_SI_NO_AÑO_1: ¿Desayuna?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No desayuna", 1 = "Desayuna todos los días", 2 = "Solo Entre Semana", 3 = "Solo en Fin de Semana"

DESAYUNO_L_V_AÑO_1: Hora que realiza el desayuno entre semana. Es una variable continua, representada como date.

DESAYUNO_S_D_AÑO_1: Hora que realiza el desayuno en fin de semana. Es una variable continua, representada como date.

ALMUERZO_L_V_AÑO_1: Hora que realiza el almuerzo entre semana. Es una variable continua, representada como date.

ALMUERZO_S_D_AÑO_1: Hora que realiza el almuerzo en fin de semana. Es una variable continua, representada como date.

CENA_L_V_AÑO_1: Hora en la que cena entre semana. Es una variable continua, representada como date.

CENA_S_D_AÑO_1: Hora en la que cena en fin de semana. Es una variable continua, representada como date.

VER_SENTADO_TV_VIDEO_L_V_AÑO_1: Horas que ve la televisión sentado al día entre semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

VER_SENTADO_TV_VIDEO_S_D_AÑO_1: Horas que ve la televisión sentado al día en fin de semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

VER_SENTADO_PC_MÓVIL_L_V_AÑO_1: Horas que está sentado con el ordenador o móvil al día entre semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

VER_SENTADO_PC_MÓVIL_S_D_AÑO_1: Horas que está sentado con el ordenador o móvil al día en fin de semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

DESPLAZAMIENTO_SENTADO_L_V_AÑO_1: Horas que está sentado durante el desplazamiento al trabajo u ocio al día entre semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

DESPLAZAMIENTO_SENTADO_S_D_AÑO_1: Horas que está sentado durante el desplazamiento al trabajo u ocio al día en fin de semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

SENTADO_TOTAL_L_V_AÑO_1: Horas que está sentado en total al día entre semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes

categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

SENTADO_TOTAL_S_D_AÑO_1: Horas que está sentado en total al día en fin de semana. Es una variable categórica **ordinal**, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Nunca", 1 = "< 30 minutos", 2 = "30–60 minutos", 3 = "1 hora",4 = "2 horas",5 = "3 horas",6 = "4 horas",7 = "5 horas",8 = "6 horas",9 = "7 horas",10 = "8 horas",11 = "9 horas o más"

CARNE_ROJA_PROCESADA_SEMANA_AÑO1: Cantidad carne procesada semana. Es una variable continua, representada como numeric.

BEBIDAS_CARBONATADAS_DIA_AÑO1: Litros al dia bebidas carbonatadas. Es una variable continua, representada como numeric.

MEDAS_CORREGIDO_1_AÑO1: ¿Usa usted el aceite de oliva principalmente para cocina?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si".

MEDAS_CORREGIDO_2_AÑO1: ¿Cuánto aceite de oliva consume en total al día (incluyendo el usado para freír, el de las comidas fuera de casa, las ensaladas, etc.)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 4 cucharadas al día", 1 = "≥ 4 cucharadas al día"

MEDAS_CORREGIDO_3_AÑO1: ¿Cuántas raciones de verduras u hortalizas consume al día (las guarniciones acompañamientos contabilizan como ½ ración)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 2 al día", 1 = "≥ 2 al día (al menos una en ensalada o cruda)"

MEDAS_CORREGIDO_4_AÑO1: ¿Cuántas piezas de fruta (incluyendo el zumo natural) consume al día?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 al día",1 = "≥ 3 al día"

MEDAS_CORREGIDO_5_AÑO1: ¿Cuántas raciones de carnes rojas, hamburguesas, salchichas o embutidos consume al día (una ración equivale a 100-150 gr.)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "≥ 1 al día", 1 = "< 1 al día"

MEDAS_CORREGIDO_6_AÑO1: ¿Cuántas raciones de mantequilla, margarina o nata consume al día (porción individual equivale a 12 gr.)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "≥ 1 al día", 1 = "< 1 al día"

MEDAS_CORREGIDO_7_AÑO1: ¿Cuántas bebidas carbonatadas y/o azucaradas consume al día?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "≥ 1 al día", 1 = "< 1 al día"

MEDAS_CORREGIDO_8_AÑO1: ¿Bebe vino? ¿Cuánto consume a la semana?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 7 vasos a la semana", 1 = "≥ 7 vasos a la semana"

MEDAS_CORREGIDO_9_AÑO1: ¿Cuántas raciones de legumbres consume a la semana (una ración o plato equivale a 150 gr.)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 a la semana", 1 = "≥ 3 a la semana"

MEDAS_CORREGIDO_10_AÑO1: ¿Cuántas raciones de pescado o marisco consume a la semana (un plato, pieza o ración equivale a 100-150 gr) de pescado ó 4-5 piezas de marisco?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 a la semana", 1 = "≥ 3 a la semana"

MEDAS_CORREGIDO_11_AÑO1: ¿Cuantas veces consume repostería comercial (no casera) como galletas, flanes, dulces o pasteles a la semana?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "≥ 2 a la semana", 1 = "< 2 a la semana"

MEDAS_CORREGIDO_12_AÑO1: ¿Cuántas veces consume frutos secos a la semana (una ración equivale a 30 gr.)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 a la semana", 1 = "≥ 3 a la semana"

MEDAS_CORREGIDO_13_AÑO1: ¿Consume preferentemente carne de pollo, pavo o conejo en vez de ternera, cerdo, hamburguesa o salchichas (carne de pollo: una pieza o ración equivale a 100-150 gr)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

MEDAS_CORREGIDO_14_AÑO1: ¿Cuántas veces a la semana consume los vegetales cocinados, la pasta, el arroz, u otros platos aderezados con salsa de tomate, ajo, cebolla o puerro elaborada a fuego lento con aceite de oliva (sofrito)?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 2 a la semana", 1 = "≥ 2 a la semana"

MEDAS_CORREGIDO_TOTAL_AÑO1: Sumatorio cuestionario MEDAS. Es una variable continua, representada como numeric.

INS_1_SUFICIENTES_ALIMENTOS_AÑO_1: ¿Usted se haya preocupado por no tener suficientes alimentos para comer por falta de dinero u otros recursos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_2_ALIMENTOS_NO_SALUDABLES_AÑO_1: Pensando aún en los últimos 12 meses ¿hubo alguna vez en que usted no haya podido comer alimentos saludables y nutritivos por falta de dinero u otros recursos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_3_POCA_VARIEDAD_AÑO_1: ¿Hubo alguna vez en que usted haya comido poca variedad de alimentos por falta de dinero u otros recursos?. Es una variable categórica

nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_4_SALTO_COMIDA_AÑO_1: ¿Hubo alguna vez en que usted haya tenido que dejar de desayunar, almorzar o cenar porque no había suficiente dinero u otros recursos para obtener alimentos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_5_MENOS_COMIDA_AÑO_1: Pensando aún en los últimos 12 meses, ¿ hubo alguna vez en que usted haya comido menos de lo que pensaba que debía comer por falta de dinero u otros recursos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_6_SIN_ALIMENTOS_AÑO_1: ¿Hubo alguna vez en que su hogar se haya quedado sin alimentos por falta de dinero u otros recursos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_7_HAMBRE_AÑO_1: ¿Hubo alguna vez en que usted haya sentido hambre pero no comió porque no había suficiente dinero u otros recursos para obtener alimentos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INS_8_SIN_COMER_AÑO_1: ¿Hubo alguna vez en que usted haya dejado de comer todo un día por falta de dinero u otros recursos?. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INSEG_ALIM_FIES_AÑO1: Clasificación por Pregunta FIES Año 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Seguridad Alimentaria", 1 = "Inseguridad Alimentaria Leve (P1,P2,P3=1)", 2 = "Inseguridad Alimentaria Moderada (P4,P5,P6=1)", 3 = "Inseguridad Alimentaria Severa (P7,P8=1)"

INSEG_ALIM_FIES_MOD_SEVERE_AÑO1: Clasificación Moderada-Severa por Pregunta FIES Año 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No tiene Inseguridad Moderada–Severa", 1 = "Tiene Inseguridad Moderada–Severa"

INSEG_ALIM_FIES_SEVERE_AÑO1: Clasificación Severa por Pregunta FIES Año 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No tiene Inseguridad Severa", 1 = "Tiene Inseguridad Severa"

SCORE7_SMOKE_AÑO1: Score Life Simple 7: Hábito tabaquico Año 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SCORE7_IMC_AÑO1: Score Life Simple 7: Indice de Masa Corporal AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

PASEO_TRANQUILO_MIN_WEEK_AÑO1: Paseo tranquilo minutos semana. Es una variable continua, representada como numeric.

CAMINAR_DEPRISA_MIN_WEEK_AÑO1: Caminar deprisa minutos semana. Es una variable continua, representada como numeric.

CAMINAR_CAMPO_MIN_WEEK_AÑO1: Caminar campo minutos semana. Es una variable continua, representada como numeric.

HUERTO_CAMPO_MIN_WEEK_AÑO1: Huerto minutos semana. Es una variable continua, representada como numeric.

EJERCICIO_FISICO_MIN_WEEK_AÑO1: Ejericio fisico minutos semana. Es una variable continua, representada como numeric.

SCORE_7_MODERADA_MIN_WEEK_AÑO_1: Actividad fisica moderada minutos semana. Es una variable continua, representada como numeric.

SCORE_7_VIGOROSA_MIN_WEEK_AÑO_1: Actividad fisica vigorosa minutos semana. Es una variable continua, representada como numeric.

SCORE_7_SUMA_MODERADA_VIGOROSA_MIN_AÑO_1: Actividad fisica moderada + vigorosa minutos semana. Es una variable continua, representada como numeric.

SCORE7_PHYSICALACTIVITY_MIN_AÑO1: Clasificación actividad física según minutos AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SUMA_FRUTAS_VERDURA_AÑO1: nan. Es una variable categórica nominal, representada como numeric.

COMPONENTE_DIETA_VEGETAL_AÑO1: Componente Score Dieta saludable: Frutas y vegetales AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 4,5 al día", 1 = "mayor o igual 4,5 al día"

COMPONENTE_DIETA_PESCADO_AÑO1: Componente Score Dieta saludable: Pescado AÑO 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 2 a la semana", 1 = "mayor o igual 2 a la semana"

COMPONENTE_DIETA_FIBRA_AÑO1: Componente Score Dieta saludable: Fibra AÑO 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "< 3 al día", 1 = "mayor o igual a 3 al día"

COMPONENTE_DIETA_SAL_AÑO1: Componente Score Dieta saludable: Sal AÑO 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes

categorías o valores posibles: 0 = "Preguntas al < 3", 1 = "Preguntas sal = 3". Esta calsificación viene de un calculo de 3 preguntas de sal de arriba.

COMPONENTE_DIETA_AZUCAR_AÑO1: Componente Score Dieta saludable: Bebidas azucaradas AÑO 1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "> 3 latas (0,99 litros)", 1 = "menor o igual 3 latas (0,99 litros)"

SUMA_COMPONENTES_DIETA_AÑO1: nan. Es una variable categórica nominal, representada como numeric.

SCORE7_DIETA_AÑO1: Score Life Simple 7: Dieta saludable AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SCORE7_COLESTEROL_AÑO1: Score Life Simple 7: Colesterol total AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SCORE7_PRESION_AÑO1: Score Life Simple 7: Presión arterial AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SCORE7_GLUCOSA_AÑO1: Score Life Simple 7: Glucosa en ayunas AÑO 1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Pobre", 1 = "Intermedia", 2 = "Ideal"

SCORE7_SMOKE_IDEAL_AÑO1: Ideal Score 7 Smoke AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_BMI_IDEAL_AÑO1: Ideal Score 7 BMI AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_PHYSICAL_ACTIVITY_IDEAL_AÑO1: Ideal Score 7 Physical activity AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_HEALTHY_DIET_IDEAL_AÑO1: Ideal Score 7 Healthy diet AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_TOTAL_CHOLESTEROL_IDEAL_AÑO1: Ideal Score 7 Total Cholesterol AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_BLOOD_PRESSURE_IDEAL_AÑO1: Ideal Score 7 Blood Pressure AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_FASTING_GLUCOSE_IDEAL_AÑO1: Ideal Score 7 Fasting Plasma Glucose AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No ideal, 1= Ideal

SCORE7_TOTAL_PUNTOS_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE7_IDEAL_NO_IDEAL_AÑO1: Clasificación de la puntuación en IDEAL o NO IDEAL AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0= No cumple item en ideal, 1= Cumple item en ideal

SCORE7_PUNTOS_SUMA_14_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_DIETA_AÑO1: Calculado a partir de puntos del MEDAS CORREGIDO adultos y niños AÑO1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "0 (0–2 puntos en MEDAS)", 25 = "25 (3–5 puntos en MEDAS)", 50 = "50 (6–8 puntos en MEDAS)", 80 = "80 (9–11 puntos en MEDAS)", 100 = "100 (12–14 puntos en MEDAS)"

SCORE8_ACTIVIDAD_FISICA_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_TABACO_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_SUEÑO_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_IMC_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_NONHDL_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_GLUCOSA_AÑO1: nan. Es una variable continua, representada como numeric.

SCORE8_PRESION_ARTERIAL_AÑO1: nan. Es una variable continua, representada como numeric.

ENFERMEDAD_HTA_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_DIABETES_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_DISLIPEMIA_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_CARDIOPATIA_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_EPOC_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_HEPATOPATIAS_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_CEREBROVASCULAR_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ENFERMEDAD_CANCER_AÑO1: nan. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

HIPOLIPEMIANTE_AÑO1: Sujetos con tratamiento hipolipemiante en año1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ANTIHIPERTENSIVO_AÑO1: Sujetos con tratamiento antihipertensivo en año1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

INSULINA_AÑO1: Sujetos con tratamiento insulina en año1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

ANTIDIABETICOORAL_AÑO1: Sujetos con tratamiento antidiabetico oral en año1. Es una variable categórica nominal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No", 1 = "Si"

SCORE8_TOTAL_PUNTOS_AÑO1: Total puntos SCORE 8. Es una variable continua, representada como numeric.

SCORE8_CLASIF_CVH_AÑO1: Clasificación recomendada CVH SCORE 8 AÑO1. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 1 = "High CVH (80–100 points)", 2 = "Moderate CVH (50–79 points)", 3 = "Low CVH (0–49 points)"

PHQ9_AÑO1_1: Poco interés o placer en hacer cosas. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_2: Se ha sentido decaído(a), deprimido(a) o sin esperanzas. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_3: Ha tenido dificultad para quedarse o permanecer dormido(a), o ha dormido demasiado. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_4: Se ha sentido cansado(a) o con poca energía. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_5: Sin apetito o ha comido en exceso. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_6: Se ha sentido mal con usted mismo(a) – o que es un fracaso o que ha quedado mal con usted mismo(a) o con su familia. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_7: Ha tenido dificultad para concentrarse en ciertas actividades, tales como leer el periódico o ver la televisión. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_8: ¿Se ha movido o hablado tan lento que otras personas podrían haberlo notado? o lo contrario – muy inquieto(a) o agitado(a) que ha estado moviéndose mucho más de lo normal. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_9: Pensamientos de que estaría mejor muerto(a) o de lastimarse de alguna manera. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

PHQ9_AÑO1_TOTAL: nan. Es una variable continua, representada como numeric.

PHQ9_DIFICULTAD: ¿Qué tanta dificultad le han dado estos problemas para hacer su trabajo, encargarse de las tareas del hogar, o llevarse bien con otras personas?. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No ha sido dificil", 1 = "Un poco dificil",2 = "Muy dificil",3 = "Extremadamente dificil"

PHQ_9_AÑO1_CATEGORIAS: nan. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No depression (0 ptos)", 1 = "Minimal depression (1–4 ptos)", 2 = "Mild depression (5–9 ptos)", 3 =

"Moderate depression (10–14 ptos)", 4 = "Moderately severe depression (15–19 ptos)" 5 = "Severe depression (20–27 ptos)".

PHQ9_AÑO1_CAT_MOD_SEV: nan. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Minimal–mild depression (1–9 ptos)", 1 = "Moderate–Severe depression (10–27 ptos)"

PHQ9_AÑO1_CAT_MIN_OTHER: No depresión o mínima frente a las demás. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No depresión o mínima", 1 = "Depresión leve o superior"

GAD7_AÑO1_1: Se ha sentido nervioso(a), ansioso(a) o con los nervios de punta. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_2: No ha sido capaz de parar o controlar su preocupación. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_3: Se ha preocupado demasiado por motivos diferentes. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles:0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_4: Ha tenido dificultad para relajarse. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_5: Se ha sentido tan inquieto(a) que no ha podido quedarse quieto(a). Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o

valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_6: Se ha molestado o irritado fácilmente. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_7: Ha tenido miedo de que algo terrible fuera a pasar. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ningún día", 1 = "Varios días",2 = "Más de la mitad de los días",3 = "Casi todos los días"

GAD7_AÑO1_TOTAL: nan. Es una variable continua, representada como numeric.

GAD7_AÑO1_CATEGORIAS: nan. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 1 = "Minimal anxiety (0–4 ptos)", 2 = "Mild anxiety (5–9 ptos)", 3 = "Moderate anxiety (10–14 ptos)", 4 = "Severe anxiety (15–21 ptos)"

GAD7_AÑO1_CAT_MOD_SEV: nan. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "No moderate severe anxiety (0–9)", 1 = "Moderate–Severe anxiety (10–21 ptos)"

GAD7_AÑO1_CAT_MIN_OTHER: nan. Es una variable categórica ordinal, representada como numeric. Presenta las siguientes categorías o valores posibles: 0 = "Ansiedad minima, 1 = "Ansiedad leve o superior"

5.1.2 LIMPIEZA DE LA BASE DE DATOS

Una vez descritas las características y las distintas variables de la base de datos procedemos a hacer una limpieza de la misma, pues encontramos algunas filas y columnas incompletas o de poca utilidad y queremos solucionar este problema. Para ello, hemos desarrollado un código en Python donde realizamos los siguientes pasos:

- En primer lugar cargamos el archivo Excel y eliminamos las columnas que están claramente incompletas y que son innecesarias en nuestra base de datos a simple vista.
- En segundo lugar, en las columnas donde un valor nulo equivale a un cero en la práctica sustituimos los valores nulos por ceros.
- Acto seguido, eliminamos las filas con más de 10 valores nulos, pues podemos considerar que son pacientes a los que no se les han tomado bien los datos o se les han tomado datos insuficientes. Escogemos el número 10 no arbitrariamente, sino porque si cogemos un número menor podemos perder mucha información, mientras que eliminando aquellas personas con al menos diez valores nulos no perdemos muchas muestras mientras sí que reducimos mucho la cantidad de variables incompletas, y las que quedan tan solo carecen de como mucho doce celdas nulas como veremos a continuación. Para llegar a que 10 es el mejor número posible hemos de ejecutar el código con distintos límites y valorar los resultados nosotros mismos.
- A conticuación vamos a aprovechar este código para convertir la hora a estilo decimal para poder trabajar con estas columnas de forma más cómoda.
- Por último, guardamos la nueva tabla limpia e imprimimos los valores nulos que quedan en cada columna que tiene al menos un valor nulo.

```
import pandas as pd

def procesar_excel(ruta_archivo):
    # Cargar el archivo Excel
    df = pd.read_excel(ruta_archivo)

    # Eliminar columnas incompletas
    columnas_eliminar = ["CODIGO_ID", "ID_CLUSTER","OCUPACION_TIPO_AÑO_1",
"OCUPACION_JORNADA_AÑO_1", "MUJER_EDAD_GRASA_MUSC_AÑO1",
"HOMBRE_EDAD_GRASA_MUSC_AÑO1", "LDL_COLESTEROL_AÑO_1",
"TIPO_EJERCICIO_FISICO_AÑO_1", "SCORE8_NONHDL_AÑO1",
"MEDIA_PULSACIONES_AÑO_1"
                          "AMIGOS_FAMILIA_ENTRE_SEMANA_HORAS_AÑO_1",
                         "AMIGOS_FAMILIA_FIN_DE_SEMANA_HORAS_AÑO_1",
                         "SIESTA_TIEMPO_L_V_AÑO_1",
                         "SIESTA_TIEMPO_S_D_AÑO_1",
```

```
                        "DESAYUNO_L_V_AÑO_1",
                        "DESAYUNO_S_D_AÑO_1",
                        "ALMUERZO_L_V_AÑO_1",
                        "ALMUERZO_S_D_AÑO_1",
                        "CENA_L_V_AÑO_1",
                        "CENA_S_D_AÑO_1",
                        "AMIGOS_FAMILIA_ENTRE_SEMANA_HORAS_AÑO_1"]
    df = df.drop(columns=columnas_eliminar, errors='ignore')

    # Reemplazar valores nulos en columnas específicas por 0
    columnas_a_reemplazar = ["EXFUMADOR_AÑO_1", "FUMADOR_AÑO_1",
"BEBEDOR_TIPO_AÑO_1", "BEBEDOR_FREQ_AÑO_1",
                             "BEBEDOR_CANTIDAD_AÑO_1",
"BEBEDOR_COMIDAS_AÑO_1", "REF_MUSCULO_AÑO1",
                             "EMBARAZADA_SI_NO_AÑO1",
"DIAS_SIESTA_L_D_AÑO_1"]


    for col in columnas_a_reemplazar:
        if col in df.columns:
            df[col] = df[col].fillna(0)

    # Eliminar filas con más de 10 valores nulos
    df = df[df.isnull().sum(axis=1) <= 10]

    # Convertir las horas en formato HH:MM:SS a decimal (ej: 03:30:00 ->
3.5)
    for col in df.columns:
        if df[col].dtype == 'object':  # Solo trabajar con columnas tipo
string
            df[col] = df[col].astype(str)  # Convertir todo a string por
seguridad
            if df[col].str.match(r'^\d{1,2}:\d{2}:\d{2}$',
na=False).all():  # Verificar formato HH:MM:SS
                df[col] = pd.to_datetime(df[col], format='%H:%M:%S',
errors='coerce')
                df[col] = df[col].dt.hour + df[col].dt.minute / 60 +
df[col].dt.second / 3600  # Convertir a decimal

    num_filas = df.shape[0]

    print("Columnas con valores nulos antes de reemplazo:")
    print(df.isnull().sum()[df.isnull().sum() > 0])
```

```
    print(f"Número de filas después de la modificación: {num_filas}")

    # Contar valores nulos en toda la tabla
    nulos_totales = df.isnull().sum().sum()

    # Guardar el archivo modificado
    df.to_excel("archivo_modificado.xlsx", index=False)

    print(f"Cantidad de valores nulos en la tabla después de la
modificación: {nulos_totales}")


# Llamar a la función con el nombre del archivo Excel
procesar_excel("Base_Angel_28_3_25.xlsx")
```

Este código, como resultado, nos guarda una nueva tabla limpia y nos imprime las columnas que siguen con valores nulos:

```
Columnas con valores nulos antes de reemplazo:
ALCOHOL_AÑO_1                                1
CONSUMO_PICOTEO_AÑO_1                        1
TALLA_AÑO_1                                  4
IMC_AÑO_1                                    4
PERIMETRO_CINTURA_AÑO1                       8
GRASA_CORPORAL_AÑO_1                        12
REF_GRASA_CORPORAL_AÑO1                     12
MASA_MUSCULAR_AÑO_1                         12
MEDIA_PRESION_ARTERIAL_SISTOLICA_AÑO_1       1
MEDIA_PRESION_ARTERIAL_DIASTOLICA_AÑO_1      1
MEDIA_PULSACIONES_AÑO_1                     23
NON_HDL_COLESTEROL_AÑO_1                     3
HDL_COLESTEROL_AÑO_1                         2
SMET_OBESIDAD_ABDOMINAL_AÑO1                 8
SMET_CRITERIOS_SUMA_AÑO1                     8
SMET_AO_IDF_AÑO1                             8
SMET_BP_IDF_AÑO1                             1
SIND_IDF_AÑO1                                8
WHtR_AÑO1                                   12
WHrR_Clasif_AÑO1                             8
INDICE_CT_HDL_AÑO_1                          4
INDICE_CT_HDL_CLASIFICACION_AÑO_1            4
PASEO_TRANQUILO_DIAS_AÑO_1                   2
PASEO_TRANQUILO_MINUTOS_AÑO_1                2
SUBIR_ESCALERAS_PISOS_AÑO_1                  6
```

```
DESAYUNA_SI_NO_AÑO_1                    5
VER_SENTADO_TV_VIDEO_L_V_AÑO_1          2
VER_SENTADO_PC_MÓVIL_L_V_AÑO_1          1
DESPLAZAMIENTO_SENTADO_L_V_AÑO_1        1
DESPLAZAMIENTO_SENTADO_S_D_AÑO_1        2
SENTADO_TOTAL_L_V_AÑO_1                 1
INS_1_SUFICIENTES_ALIMENTOS_AÑO_1       1
SCORE7_PRESION_AÑO1                     1
SCORE7_BLOOD_PRESSURE_IDEAL_AÑO1        1
SCORE8_IMC_AÑO1                         3
SCORE8_PRESION_ARTERIAL_AÑO1            1
PHQ9_DIFICULTAD                         8
PHQ9_AÑO1_CAT_MOD_SEV                   6
PHQ9_AÑO1_CAT_MIN_OTHER                 1
GAD7_AÑO1_CATEGORIAS                    1
GAD7_AÑO1_CAT_MOD_SEV                   3
GAD7_AÑO1_CAT_MIN_OTHER                 2
dtype: int64
Número de filas después de la modificación: 419
Cantidad de valores nulos en la tabla después de la modificación: 195
```

Como podemos observar no hemos perdido muchas filas en nuestra base de datos y ahora tenemos muy pocas columnas (en comparación con el total) que tienen valores nulos, que al ser pocos en cada columna lo podemos resolver sustituyendo por ceros, media o moda como explicaremos a continuación.

5.1.2.1 Tratamiento de valores perdidos (missing values)

Para asegurar la completitud del conjunto de datos y minimizar la pérdida de información debido a valores perdidos (*missing values*), se aplicaron diferentes métodos de imputación en función de la naturaleza de cada variable, basados en su tipo (categórica o continua) y su interpretación específica según la documentación disponible. A continuación se detalla la estrategia adoptada para cada grupo de variables:

5.1.2.1.1 Imputación con la Moda (Variables Categóricas o Discretas)

Las variables categóricas o discretas representan clasificaciones clínicas, respuestas ordinales o estados binarios con significado categórico. La imputación mediante la moda (valor más frecuente) permite conservar la estructura de clases sin introducir sesgos cuantitativos.

Se imputaron por moda las siguientes variables:

```
columnas_moda = [
        "ALCOHOL_AÑO_1",
        "CONSUMO_PICOTEO_AÑO_1",
        "REF_GRASA_CORPORAL_AÑO1",
        "MEDIA_PULSACIONES_AÑO_1",
        "SMET_OBESIDAD_ABDOMINAL_AÑO1",
        "SMET_CRITERIOS_SUMA_AÑO1",
        "SMET_AO_IDF_AÑO1",
        "SMET_BP_IDF_AÑO1",
        "SIND_IDF_AÑO1",
        "WHrR_Clasif_AÑO1",
        "INDICE_CT_HDL_CLASIFICACION_AÑO_1",
        "DESAYUNA_SI_NO_AÑO_1",
        "INS_1_SUFICIENTES_ALIMENTOS_AÑO_1",
        "PHQ9_AÑO1_CAT_MOD_SEV",
        "PHQ9_AÑO1_CAT_MIN_OTHER",
        "PHQ9_DIFICULTAD",
        "GAD7_AÑO1_CATEGORIAS",
        "GAD7_AÑO1_CAT_MOD_SEV",
        "GAD7_AÑO1_CAT_MIN_OTHER"
```

```
    ]
```

5.1.2.1.2 Imputación con la Media (Variables Continuas)

Para variables numéricas continuas, se optó por imputar valores perdidos con la media aritmética. Este método es apropiado cuando la distribución de la variable no presenta una asimetría significativa y el objetivo es mantener la media poblacional sin distorsionar la varianza excesivamente.

Variables imputadas con la media incluyen:

```
columnas_media = [
        "TALLA_AÑO_1",
        "IMC_AÑO_1",
        "PERIMETRO_CINTURA_AÑO1",
        "GRASA_CORPORAL_AÑO_1",
        "MASA_MUSCULAR_AÑO_1",
        "MEDIA_PRESION_ARTERIAL_SISTOLICA_AÑO_1",
        "MEDIA_PRESION_ARTERIAL_DIASTOLICA_AÑO_1",
        "NON_HDL_COLESTEROL_AÑO_1",
        "HDL_COLESTEROL_AÑO_1",
        "WHtR_AÑO1",
        "INDICE_CT_HDL_AÑO_1",
        "PASEO_TRANQUILO_DIAS_AÑO_1",
        "PASEO_TRANQUILO_MINUTOS_AÑO_1",
        "SUBIR_ESCALERAS_PISOS_AÑO_1",
        "SENTADO_TOTAL_L_V_AÑO_1",
        "VER_SENTADO_TV_VIDEO_L_V_AÑO_1",
        "VER_SENTADO_PC_MÓVIL_L_V_AÑO_1",
        "DESPLAZAMIENTO_SENTADO_L_V_AÑO_1",
        "DESPLAZAMIENTO_SENTADO_S_D_AÑO_1"
    ]
```

5.1.2.1.3 Imputación con Cero (Variables Binarias o Indicadores de Ausencia)

Algunas variables binarias o indicadores clínicos donde un valor nulo probablemente indica ausencia de la condición, fueron imputadas con cero para representar la no presencia o el estado "negativo".

Se imputaron con cero las siguientes variables:

```
    columnas_cero = [
        "SCORE7_BLOOD_PRESSURE_IDEAL_AÑO1",
        "SCORE8_IMC_AÑO1",
        "SCORE8_PRESION_ARTERIAL_AÑO1",
        "SCORE7_PRESION_AÑO1"
    ]
```

5.1.2.1.4 Código

Para tratar estos valores perdidos como hemos indicado se ha hecho uso del siguiente código:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

def procesar_excel(ruta_archivo):
    # Cargar el archivo Excel
    df = pd.read_excel(ruta_archivo)

    # Eliminar columnas conflictivas
    columnas_eliminar = ['FECHA_EDAD_AÑO1', 'FECHA_NACIMIENTO',
'DORMIR_S_D_AÑO_1', 'DESPERTAR_S_D_AÑO_1']
    df = df.drop(columns=columnas_eliminar, errors='ignore')

    # === 1. Rellenar con 0 (ausencia de la condición, score binario, etc.)
===
    columnas_cero = [
        "SCORE7_BLOOD_PRESSURE_IDEAL_AÑO1",
        "SCORE8_IMC_AÑO1",
        "SCORE8_PRESION_ARTERIAL_AÑO1",
        "SCORE7_PRESION_AÑO1"
    ]
    for col in columnas_cero:
        if col in df.columns:
            df[col] = df[col].fillna(0)

    # === 2. Rellenar con la media (variables continuas) ===
    columnas_media = [
        "TALLA_AÑO_1",
        "IMC_AÑO_1",
        "PERIMETRO_CINTURA_AÑO1",
```

```
        "GRASA_CORPORAL_AÑO_1",
        "MASA_MUSCULAR_AÑO_1",
        "MEDIA_PRESION_ARTERIAL_SISTOLICA_AÑO_1",
        "MEDIA_PRESION_ARTERIAL_DIASTOLICA_AÑO_1",
        "NON_HDL_COLESTEROL_AÑO_1",
        "HDL_COLESTEROL_AÑO_1",
        "WHtR_AÑO1",
        "INDICE_CT_HDL_AÑO_1",
        "PASEO_TRANQUILO_DIAS_AÑO_1",
        "PASEO_TRANQUILO_MINUTOS_AÑO_1",
        "SUBIR_ESCALERAS_PISOS_AÑO_1",
        "SENTADO_TOTAL_L_V_AÑO_1",
        "VER_SENTADO_TV_VIDEO_L_V_AÑO_1",
        "VER_SENTADO_PC_MÓVIL_L_V_AÑO_1",
        "DESPLAZAMIENTO_SENTADO_L_V_AÑO_1",
        "DESPLAZAMIENTO_SENTADO_S_D_AÑO_1"
    ]
    for col in columnas_media:
        if col in df.columns:
            df[col] = df[col].fillna(df[col].mean())

    # === 3. Rellenar con la moda (variables categóricas) ===
    columnas_moda = [
        "ALCOHOL_AÑO_1",
        "CONSUMO_PICOTEO_AÑO_1",
        "REF_GRASA_CORPORAL_AÑO1",
        "MEDIA_PULSACIONES_AÑO_1",
        "SMET_OBESIDAD_ABDOMINAL_AÑO1",
        "SMET_CRITERIOS_SUMA_AÑO1",
        "SMET_AO_IDF_AÑO1",
        "SMET_BP_IDF_AÑO1",
        "SIND_IDF_AÑO1",
        "WHrR_Clasif_AÑO1",
        "INDICE_CT_HDL_CLASIFICACION_AÑO_1",
        "DESAYUNA_SI_NO_AÑO_1",
        "INS_1_SUFICIENTES_ALIMENTOS_AÑO_1",
        "PHQ9_AÑO1_CAT_MOD_SEV",
        "PHQ9_AÑO1_CAT_MIN_OTHER",
        "PHQ9_DIFICULTAD",
        "GAD7_AÑO1_CATEGORIAS",
        "GAD7_AÑO1_CAT_MOD_SEV",
        "GAD7_AÑO1_CAT_MIN_OTHER"
    ]
```

```
    for col in columnas_moda:
        if col in df.columns:
            df[col] = df[col].fillna(df[col].mode()[0])


    variables_dummies = [
    'GRUPO_INTERVENCION', 'TABACO_AÑO_1', 'ALCOHOL_AÑO_1',
    'BEBEDOR_TIPO_AÑO_1', 'BEBEDOR_FREQ_AÑO_1',
    'EDUCACION_AÑO_1', 'OCUPACION_AÑO_1', 'INGRESOS_AÑO_1',
    'DESAYUNA_SI_NO_AÑO_1', 'SCORE7_SMOKE_AÑO1', 'SCORE7_IMC_AÑO1',
    'SCORE7_PHYSICALACTIVITY_MIN_AÑO1', 'SCORE7_DIETA_AÑO1',
    'SCORE7_COLESTEROL_AÑO1', 'SCORE7_PRESION_AÑO1',
    'SCORE7_GLUCOSA_AÑO1', 'SCORE8_DIETA_AÑO1',
    'SCORE8_CLASIF_CVH_AÑO1', 'PHQ9_AÑO1_1', 'PHQ9_AÑO1_2',
    'PHQ9_AÑO1_3', 'PHQ9_AÑO1_4', 'PHQ9_AÑO1_5', 'PHQ9_AÑO1_6',
    'PHQ9_AÑO1_7', 'PHQ9_AÑO1_8', 'PHQ9_AÑO1_9',
    'PHQ9_DIFICULTAD',
    'GAD7_AÑO1_1', 'GAD7_AÑO1_2', 'GAD7_AÑO1_3',
    'GAD7_AÑO1_4', 'GAD7_AÑO1_5', 'GAD7_AÑO1_6', 'GAD7_AÑO1_7',
    'VER_SENTADO_TV_VIDEO_L_V_AÑO_1',
    'VER_SENTADO_TV_VIDEO_S_D_AÑO_1', 'VER_SENTADO_PC_MÓVIL_L_V_AÑO_1',
    'VER_SENTADO_PC_MÓVIL_S_D_AÑO_1', 'DESPLAZAMIENTO_SENTADO_L_V_AÑO_1',
    'DESPLAZAMIENTO_SENTADO_S_D_AÑO_1', 'SENTADO_TOTAL_L_V_AÑO_1',
    'SENTADO_TOTAL_S_D_AÑO_1'
]

# Creamos dummies
    df = pd.get_dummies(df, columns=variables_dummies, drop_first=False)

    # Mostrar conteo total de nulos después de imputación
    nulos_totales = df.isnull().sum().sum()
    num_filas = df.shape[0]

    # Guardar el archivo modificado
    df.to_excel("archivo_modificado.xlsx", index=False)

    print(f"Archivo guardado como 'archivo_modificado.xlsx'")
    print(f"Valores nulos restantes en el DataFrame: {nulos_totales}")
    print(f"Número total de filas: {num_filas}")
    print(df.isnull().sum()[df.isnull().sum() > 0])

# Ejecutar la función con el archivo
procesar_excel("archivo_modificado.xlsx")
```

Este código nos da como resultado el número de valores nulos y de filas después de la modificación:

```
Archivo guardado como 'archivo_modificado.xlsx'
Valores nulos restantes en el DataFrame: 0
Número total de filas: 419
Series([], dtype: int64)
```

Como podemos observar, el resultado es exitoso pues conseguimos reducir la cantidad de valores nulos en la tabla a 0.

Además, en este código creamos las dummies para las variables categóricas, una vez solucionado el problema de los valores nulos.

5.1.2.1.5 Conclusión

La imputación fue diseñada para preservar la integridad estadística de las variables respetando su naturaleza intrínseca. Esta estrategia mixta permite un análisis posterior más robusto, evitando tanto la eliminación de registros incompletos como la introducción de sesgos sistemáticos.

5.2 DESARROLLO DE LOS MODELOS DE REGRESIÓN: PREDICCIÓN DE VARIABLES CONTINUAS

En esta etapa crearemos los distintos modelos de predicción con Python y analizaremos en profundidad el comportamiento y predicción de cada uno de ellos tanto para la variable del estrés como la de la depresión. Además, no solo haremos una predicción de las variables de la depresión y del estrés tratándolas como variables continuas, sino que también lo haremos como variables categóricas.

5.2.1 DEFINICIONES

Antes de comenzar con el análisis de cada uno de los modelos que probaremos en el proyecto, en este apartado mostraremos las definiciones necesarias para comprender los resultados obtenidos. En el código ejecutado pediremos los siguientes Indicadores de Evaluación utilizados:

- **Accuracy (Precisión global)**: Esta métrica indica la proporción de predicciones correctas realizadas por el modelo sobre el total de casos. Es especialmente útil cuando las clases están equilibradas, ya que ofrece una visión general del rendimiento. Sin embargo, puede ser engañosa en escenarios con clases desbalanceadas, ya que un modelo puede mostrar una alta accuracy simplemente por predecir mayoritariamente la clase más común.
- **F1-score**: El F1-score es la media armónica entre la precisión (proporción de verdaderos positivos sobre los predichos como positivos) y la exhaustividad o sensibilidad (proporción de verdaderos positivos sobre los casos realmente positivos). Esta métrica es especialmente útil en contextos donde es importante equilibrar los falsos positivos y los falsos negativos, como ocurre frecuentemente en salud mental, donde una predicción errónea puede tener consecuencias clínicas importantes.
- **Matriz de confusión**: La matriz de confusión proporciona un desglose detallado de las predicciones realizadas por el modelo en comparación con los valores reales.

Permite identificar cuántos casos han sido clasificados correctamente y cuántos han sido confundidos entre clases. Es fundamental para detectar patrones de error y ajustar el modelo en función de los tipos de equivocaciones que comete.

- **Importancia de características (Feature Importance)**: Esta técnica permite identificar qué variables tienen un mayor peso o influencia en las decisiones del modelo. A través de distintas metodologías (como el análisis de ganancia en árboles de decisión o el descenso del rendimiento al eliminar variables), se puede valorar qué factores son más relevantes para la predicción, lo cual es especialmente útil para interpretación clínica y selección de variables.
- **Valores SHAP (SHapley Additive exPlanations)**: Los valores SHAP son una herramienta avanzada de interpretabilidad que permite explicar el impacto de cada variable en la predicción individual de un modelo. Basados en teoría de juegos, los SHAP values descomponen la salida del modelo en contribuciones atribuibles a cada característica, lo que facilita comprender por qué el modelo ha tomado una decisión determinada para cada paciente.

- **R^2 (coeficiente de determinación):** es una medida estadística que indica qué proporción de la variabilidad total de la variable dependiente puede ser explicada por el modelo. Su valor oscila entre 0 y 1 (aunque en algunos casos puede ser negativo si el modelo es peor que una simple media), donde 1 representa una predicción perfecta y 0 indica que el modelo no explica ninguna variabilidad. En otras palabras, R^2 permite evaluar qué tan bien se ajusta el modelo a los datos reales: un valor alto sugiere que las predicciones se acercan a los valores observados, mientras que un valor bajo indica un mal ajuste. Es una métrica especialmente útil en problemas de regresión.

- **Error Absoluto Medio (MAE - Mean Absolute Error):** es una métrica utilizada principalmente en problemas de regresión. Mide el promedio de las diferencias absolutas entre los valores reales y los valores predichos. Es decir, representa en promedio cuánto se desvía el modelo en sus predicciones respecto a los valores reales, sin penalizar de forma extra los errores más grandes.

$$MAE = (1 / n) * \Sigma_{i}^{n} |y_i - \hat{y}_i|$$

donde:

- y_i es el valor real,
- $\hat{y}_i$ es el valor predicho por el modelo,
- n es el número total de observaciones.

El MAE es fácil de interpretar: por ejemplo, un MAE de 2 unidades significa que, en promedio, el modelo se equivoca por 2 unidades en sus predicciones.

- **Error Cuadrático Medio (MSE - Mean Squared Error):** se utiliza también en regresión y mide el promedio de los errores al cuadrado entre las predicciones y los valores reales. A diferencia del MAE, el MSE penaliza más fuertemente los errores grandes, ya que eleva al cuadrado las diferencias.

$$MSE = (1 / n) * \Sigma_{i}^{n} (y_i - \hat{y}_i)^2$$

Esta métrica es útil cuando se quiere dar mayor importancia a las predicciones que tienen errores muy grandes. Sin embargo, debido a la elevación al cuadrado, su valor es menos interpretable directamente en la misma escala que los datos originales.

5.3 DESARROLLO DE LOS MODELOS DE REGRESIÓN: PREDICCIÓN DE VARIABLES CONTINUAS

La primera parte de nuestro análisis se centra en usar modelos de regresión. El objetivo aquí es muy claro: en lugar de predecir una etiqueta como "depresión leve", queremos que el modelo nos diga el número exacto de la puntuación de un cuestionario.

Decidimos empezar por aquí porque predecir el número exacto nos da una información mucho más detallada y precisa. Mientras que la clasificación mete a las personas en "cajas" o grupos la regresión nos dice exactamente en qué punto se encuentra cada persona.

Usar la regresión nos aporta dos ventajas clave:

- Ver los detalles y seguir los cambios: En lugar de solo saber la categoría, la regresión responde a la pregunta: "¿Qué puntuación exacta tiene el paciente?". Esto es súper útil para seguir de cerca su evolución y ver si un tratamiento está funcionando de verdad, punto por punto.

Detectar pequeñas mejoras: Si un paciente mejora su puntuación de 14 a 11. Para un modelo de clasificación, es posible que siga en la misma categoría, por lo que parecería que no ha habido ningún cambio. Un modelo de regresión, en cambio, sí detecta esa importante mejora de 3 puntos.

Para medir si estos modelos funcionan bien, usaremos métricas especiales para regresión (como R^2, MAE y RMSE). Nuestra estrategia será empezar con un modelo simple y luego probar otros más potentes para ver si conseguimos predicciones cada vez más precisas.

5.3.1 REGRESIÓN DE RIDGE

El punto de partida de nuestro análisis cuantitativo es la implementación de la Regresión de Ridge. Este modelo se establece como la base de referencia (baseline) fundamental, proporcionando un estándar de rendimiento objetivo. Su elección se fundamenta en ser un modelo lineal paramétrico que extiende la Regresión Lineal Ordinaria (OLS) con una mejora crucial: la regularización L2. Este mecanismo penaliza la magnitud de los coeficientes del modelo, añadiendo un término a la función de coste proporcional a la suma de los cuadrados de dichos coeficientes ($\alpha\sum\beta^2$).

El hiperparámetro clave, alpha, controla la intensidad de esta penalización. Con un alpha de cero, el modelo es idéntico a OLS. A medida que alpha aumenta, la restricción sobre los coeficientes se vuelve más severa, "encogiéndolos" hacia cero. Este proceso es una manifestación del trade-off entre sesgo y varianza: al incrementar alpha, introducimos un pequeño sesgo en el modelo, pero reducimos significativamente su varianza, haciéndolo menos sensible a las particularidades del conjunto de entrenamiento y, por tanto, más robusto. Una de sus mayores virtudes es la mitigación de la multicolinealidad. En lugar de asignar un coeficiente muy grande a una sola variable de un grupo de predictores correlacionados, Ridge distribuye el peso entre todas ellas, resultando en un modelo más estable y fiable.

La Regresión de Ridge nos da los siguientes resultados:

Métricas en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.3387, MAE: 3.6274, RMSE: 4.6698

Métricas en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.4149, MAE: 3.1910, RMSE: 4.0992

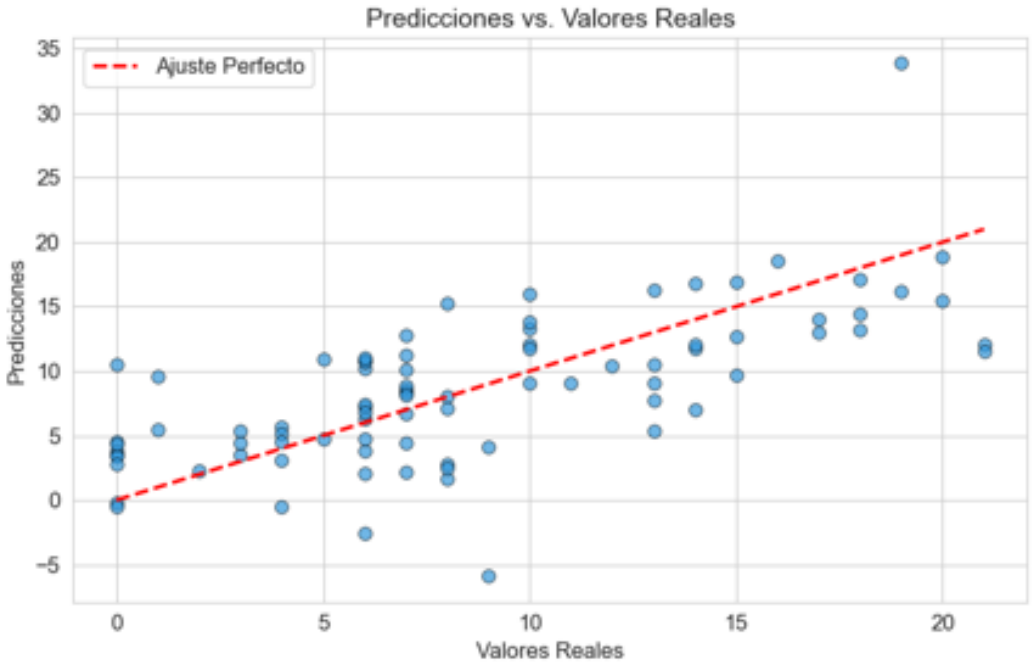


Ilustración 3 Predicciones vs. Valores Reales PHQ9 Modelo Ridge Regression

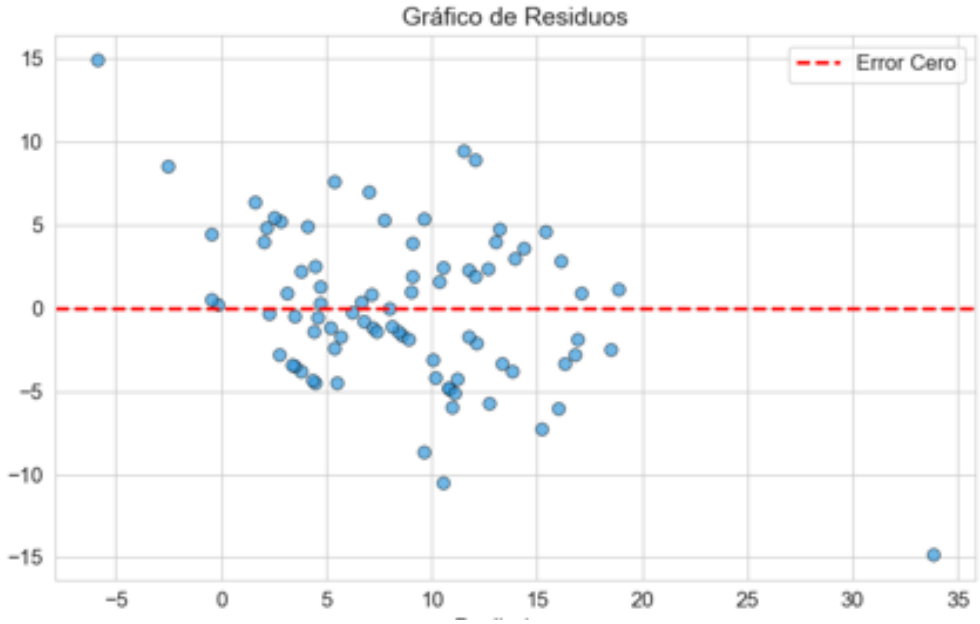


Ilustración 4 Gráfico de Residuos PHQ9 Modelo Ridge Regression

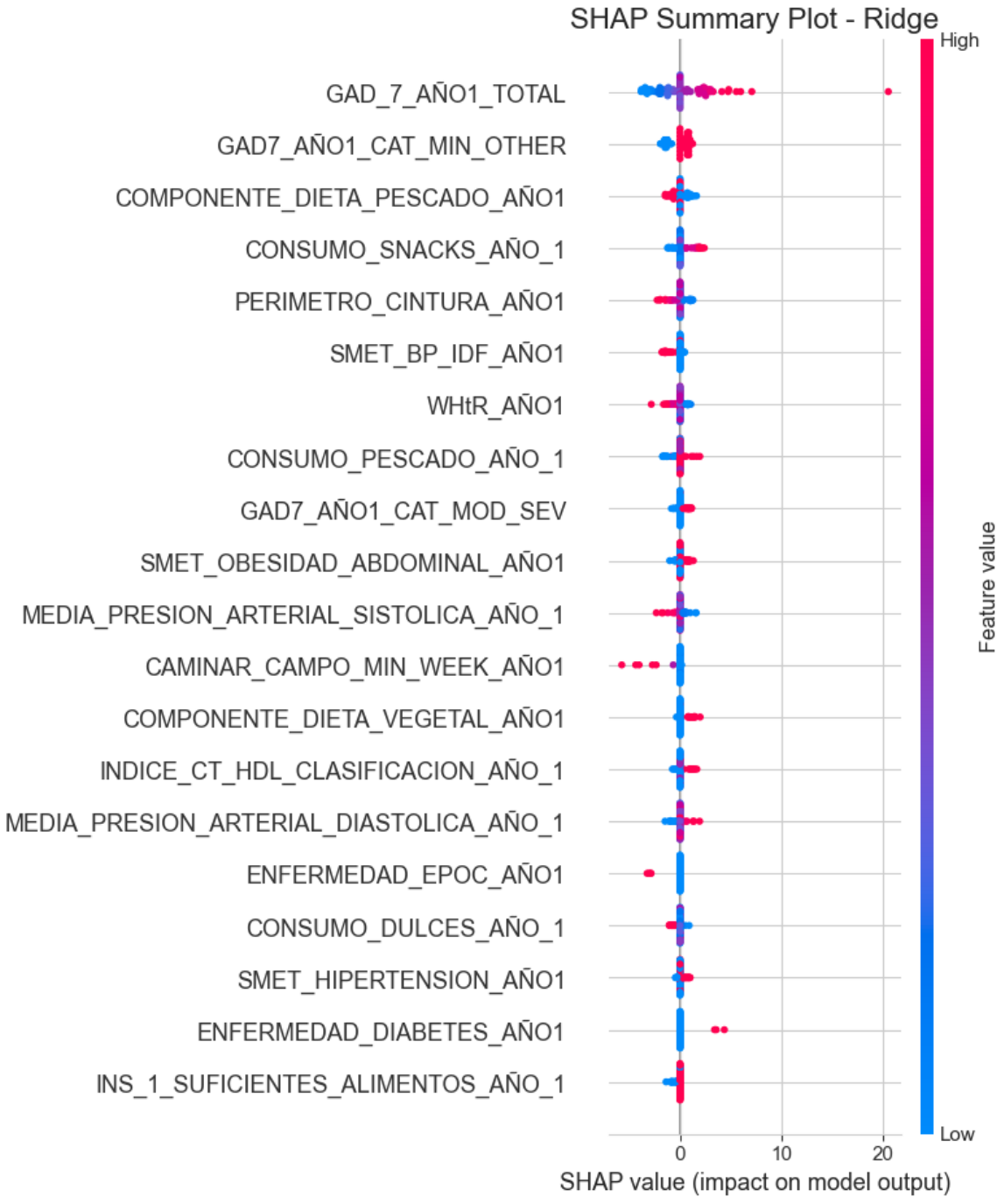


Ilustración 5 Importancia Variables PHQ9 Modelo Ridge Regression

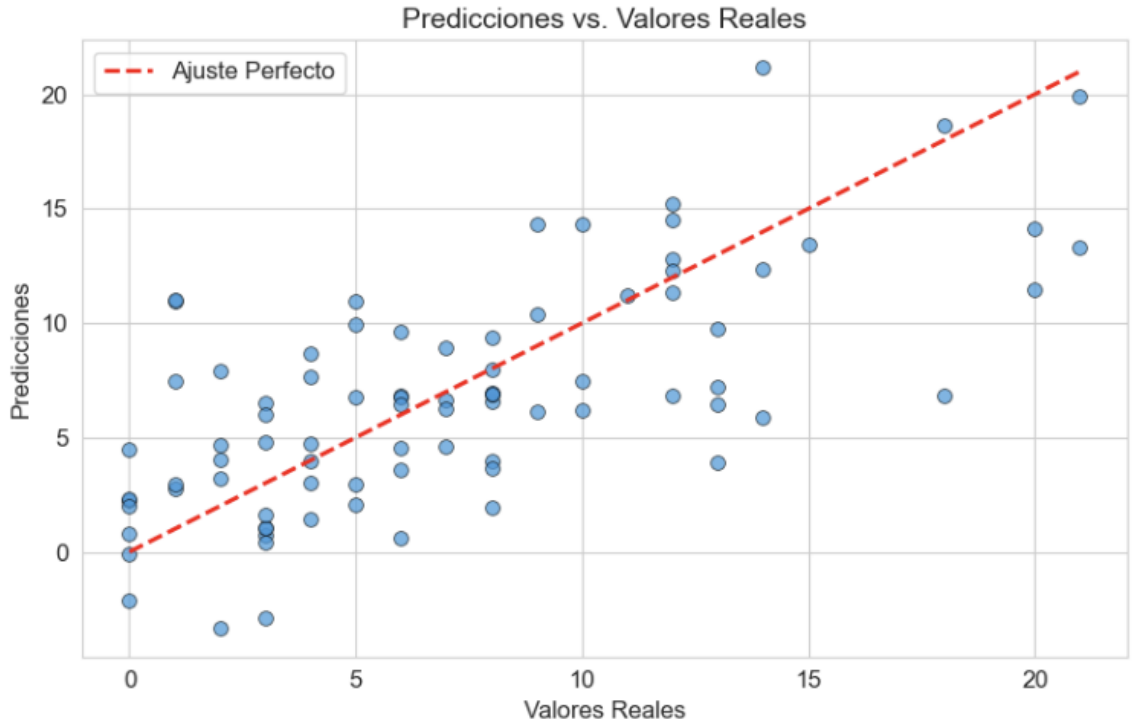


Ilustración 6 Predicciones vs. Valores Reales GAD7 Modelo Ridge Regression

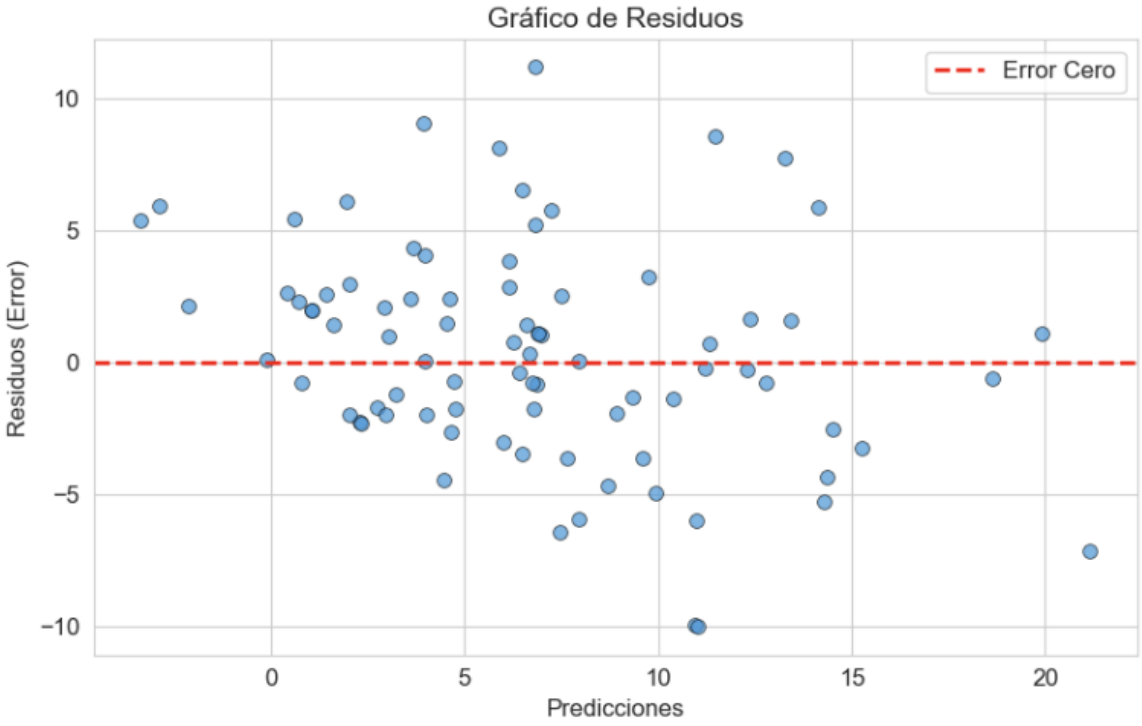


Ilustración 7 Gráfico de Residuos GAD7 Modelo Ridge Regression

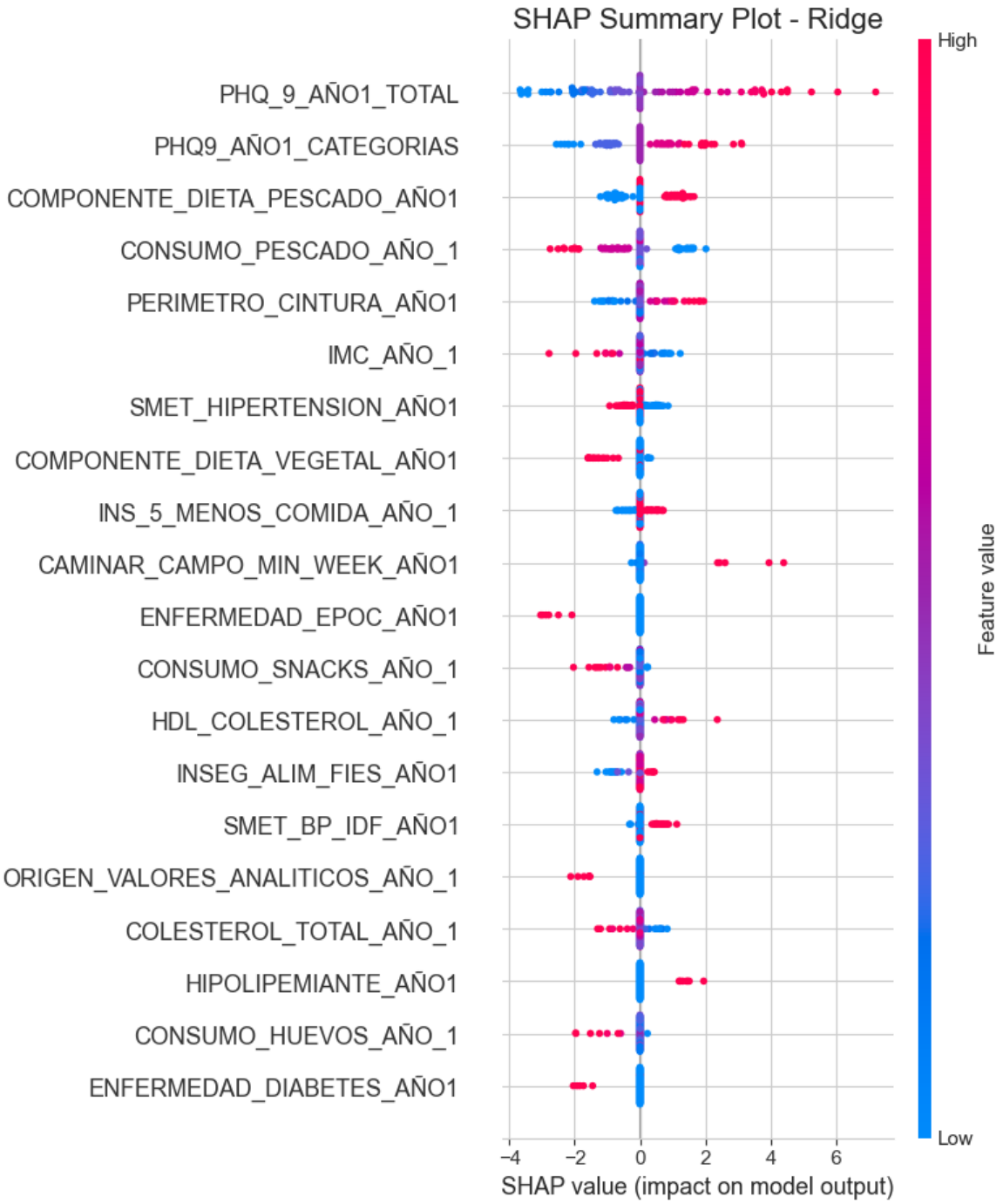


Ilustración 8 Importancia Variables GAD7 Modelo Ridge Regression

5.3.2 RANDOM FOREST REGRESSOR

Una vez establecido el rendimiento base con un modelo lineal, la metodología avanza para explorar la hipótesis de que existen relaciones no lineales complejas en los datos. Para este fin, se introduce el Random Forest Regressor, un modelo de ensamblaje no paramétrico que representa un salto cualitativo en complejidad y capacidad de modelado. Su eficacia se deriva de dos mecanismos fundamentales: bagging y la aleatorización de características.

- Bagging El modelo construye cientos de árboles de decisión independientes. Cada árbol se entrena sobre una muestra bootstrap, es decir, una muestra aleatoria con reemplazo del conjunto de datos original. Esto asegura que cada árbol aprende de una perspectiva ligeramente distinta de los datos.

- Aleatoriedad de Características: Este es el elemento que diferencia a Random Forest del bagging simple. En cada división de un nodo, el algoritmo solo considera un subconjunto aleatorio del total de características disponibles. Esta restricción es crucial porque decorrelaciona los árboles. Sin ella, la mayoría de los árboles tenderían a elegir la misma variable predictora más fuerte en las primeras divisiones, resultando en árboles muy similares entre sí.

Al promediar las predicciones de un gran número de árboles decorrelacionados, se cancela el ruido y se reduce la varianza del modelo final, logrando una predicción muy robusta. Este modelo nos proporciona los siguientes resultados:

Métricas en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.6293, MAE: 2.7237, RMSE: 3.4964

Métricas en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.5751, MAE: 2.5419, RMSE:

3.4931

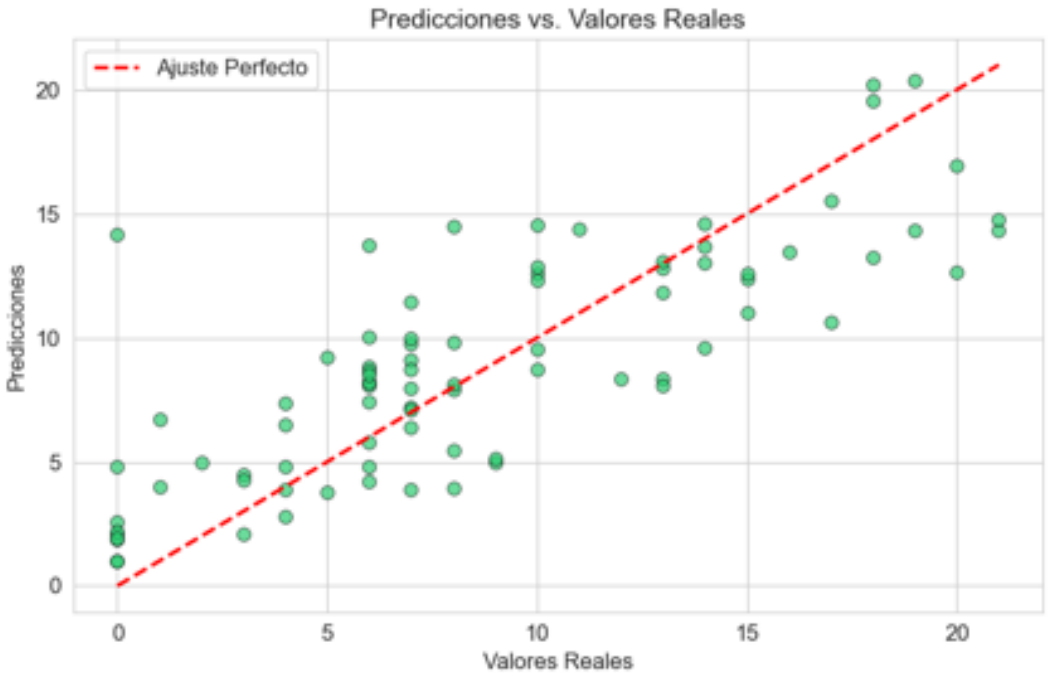


Ilustración 9 Predicciones vs. Valores Reales PHQ9 Modelo RF Regressor

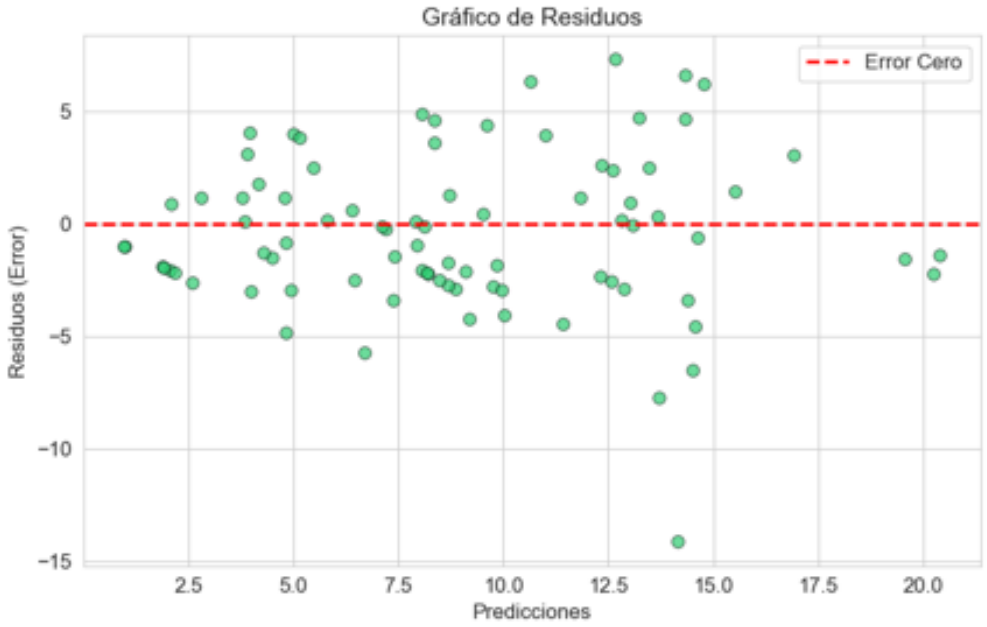


Ilustración 10 Gráfico de Residuos PHQ9 Modelo RF Regresor

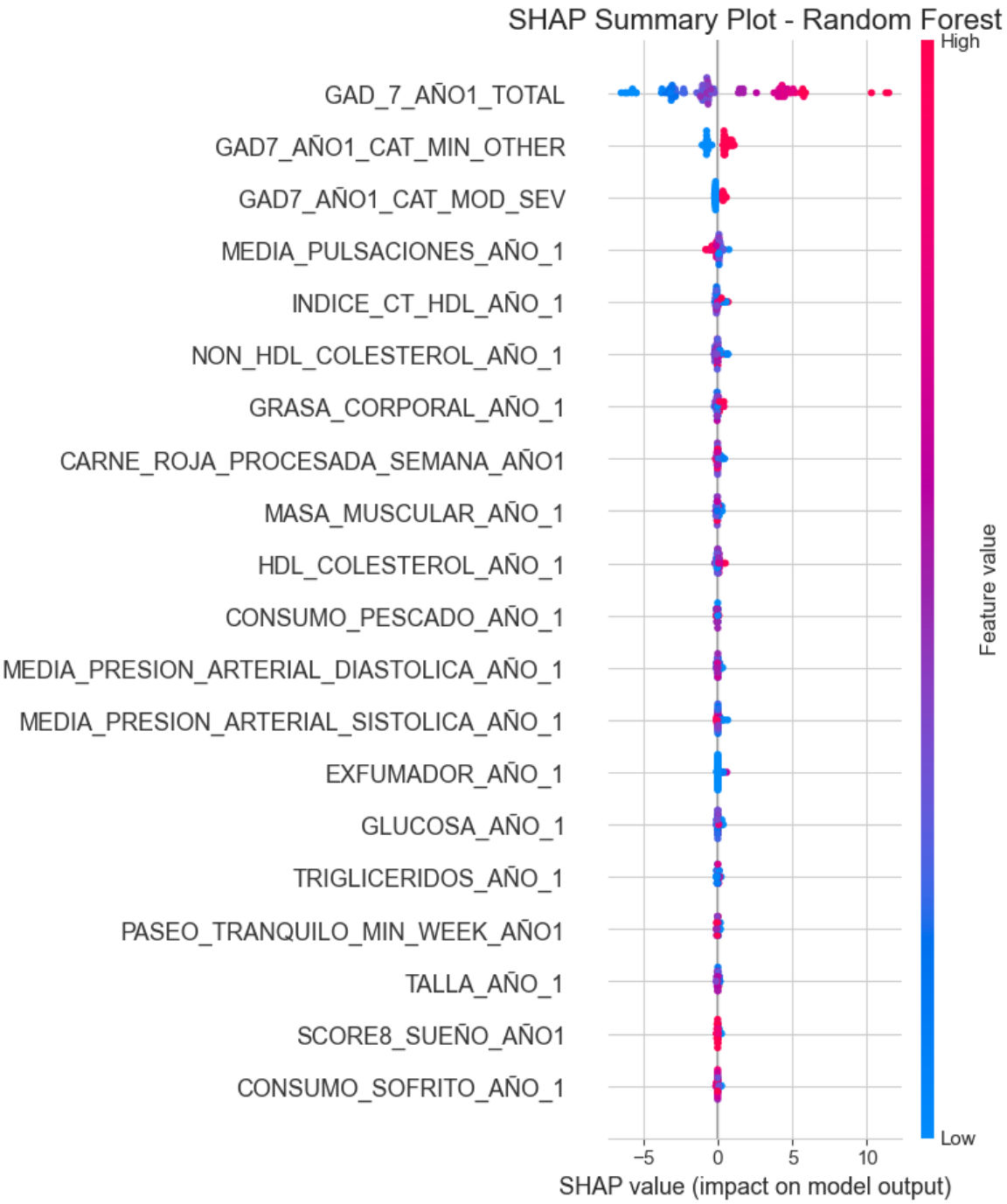


Ilustración 11 Importancia Variables PHQ9 RF Regresor

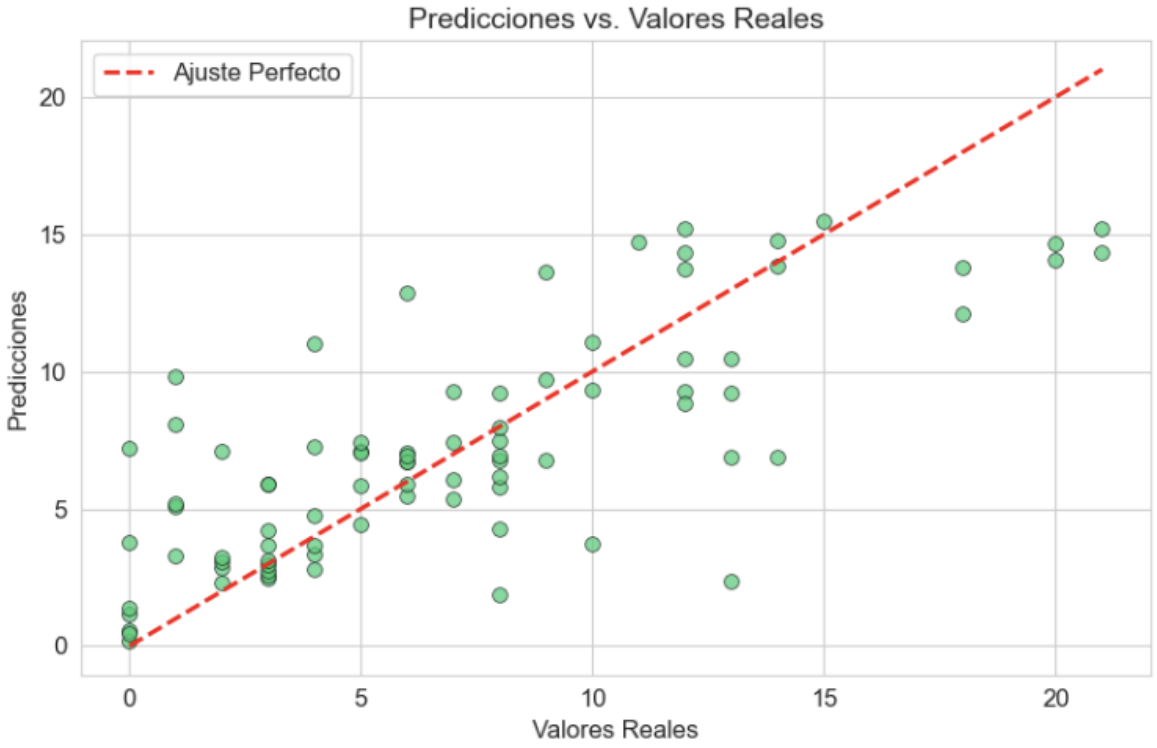


Ilustración 12 Predicciones vs. Valores Reales GAD7 Modelo RF Regresor

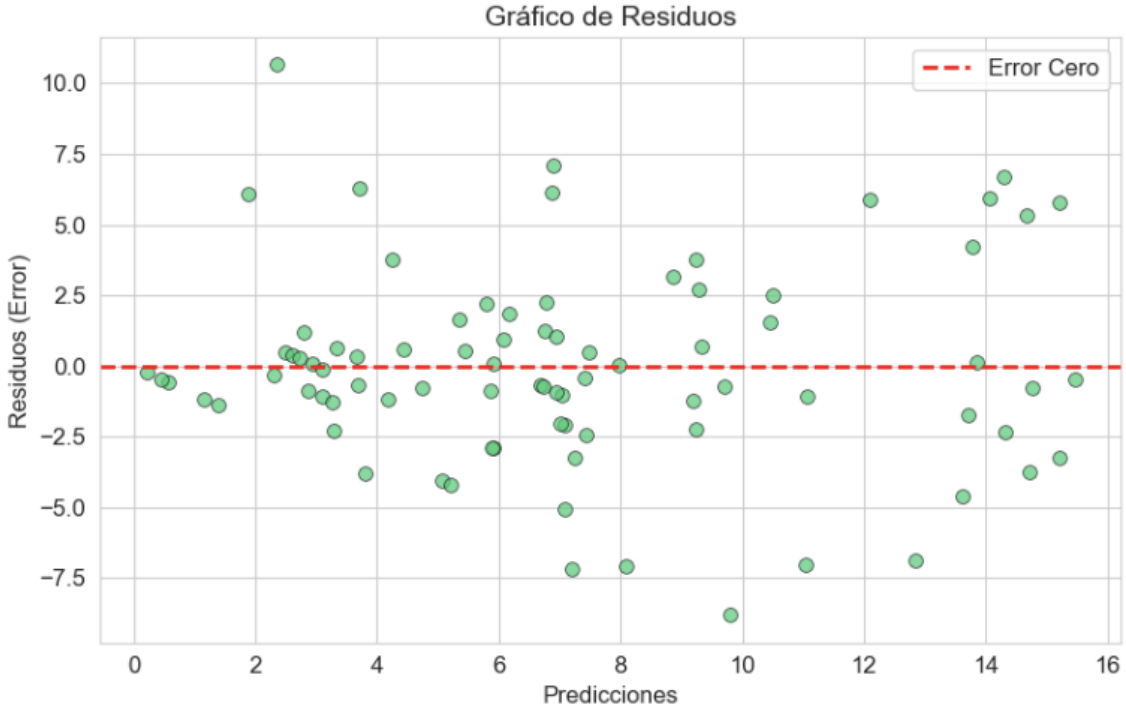


Ilustración 13 Gráfico de Residuos GAD7 Modelo RF Regressor

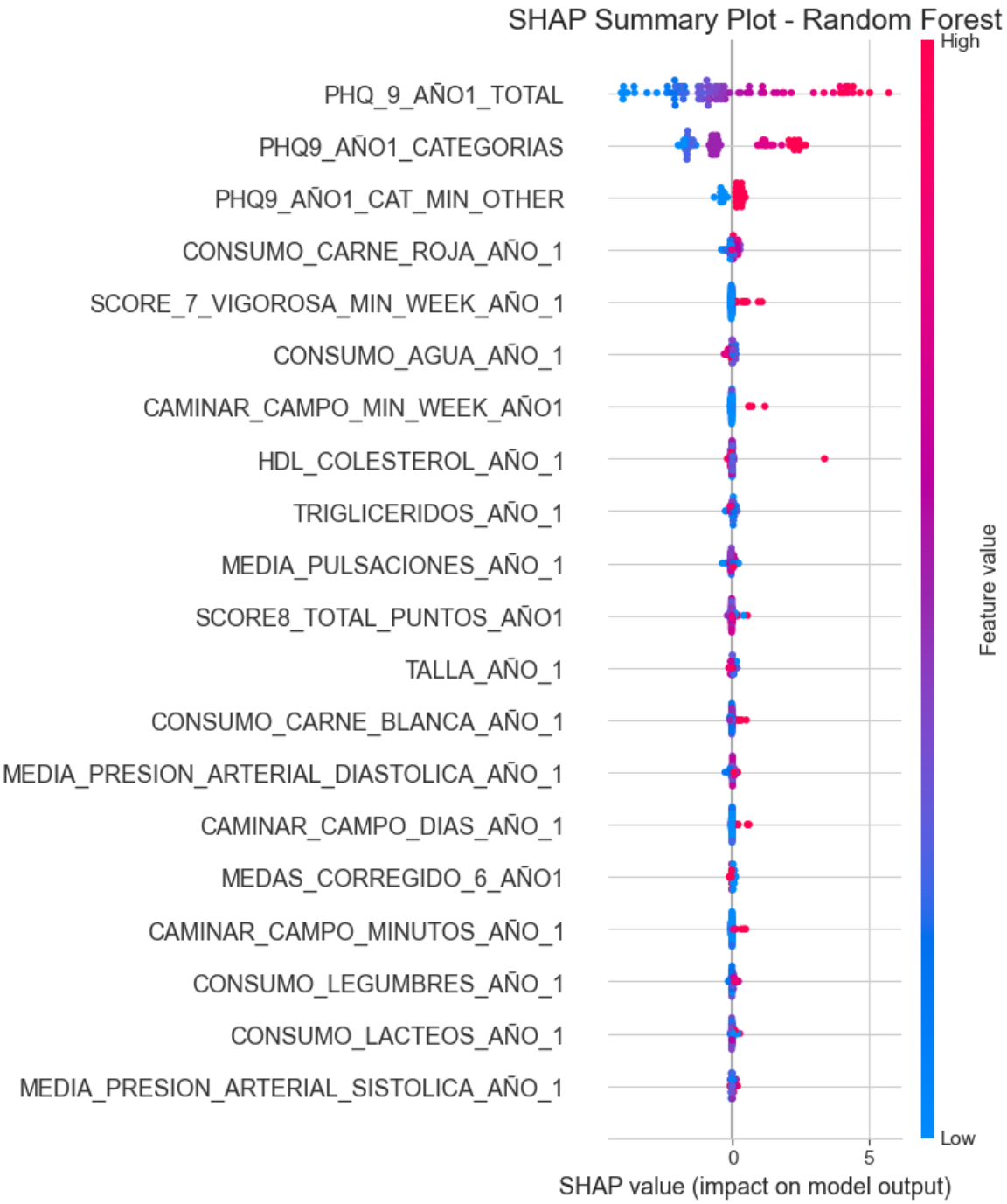


Ilustración 14 Importancia Variables PHQ9 RF Regressor

5.3.3 LIGHTGBM REGRESSOR

Habiendo evaluado un modelo de ensamblaje basado en bagging, el siguiente paso metodológico es explorar la familia de algoritmos de gradient boosting. Estos modelos construyen los árboles de forma aditiva y secuencial: cada nuevo árbol se ajusta sobre los pseudo-residuales (los errores) del ensamblaje anterior. Este enfoque permite que el modelo se concentre progresivamente en las instancias más difíciles de predecir.

El primer modelo implementado de esta categoría es LightGBM, seleccionado por su excepcional eficiencia computacional. Sus innovaciones clave incluyen:

- Algoritmo Basado en Histogramas: En lugar de evaluar cada valor único de las características continuas para encontrar el mejor punto de división (un proceso computacionalmente costoso), LightGBM agrupa los valores en un número reducido de bins y construye histogramas. Las búsquedas de divisiones se realizan sobre estos histogramas, lo que reduce drásticamente el tiempo de entrenamiento.
- Crecimiento por Hoja (Leaf-wise): A diferencia de otros algoritmos que crecen los árboles nivel por nivel, LightGBM expande el árbol desde la hoja que promete la mayor reducción en la función de pérdida. Esto le permite converger más rápidamente hacia un modelo óptimo, aunque requiere un control cuidadoso de la profundidad (max_depth) para evitar el sobreajuste en datasets pequeños.

Métricas en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.6602, MAE: 2.6131, RMSE: 3.3475

Métricas en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.5850, MAE: 2.6441, RMSE: 3.4523

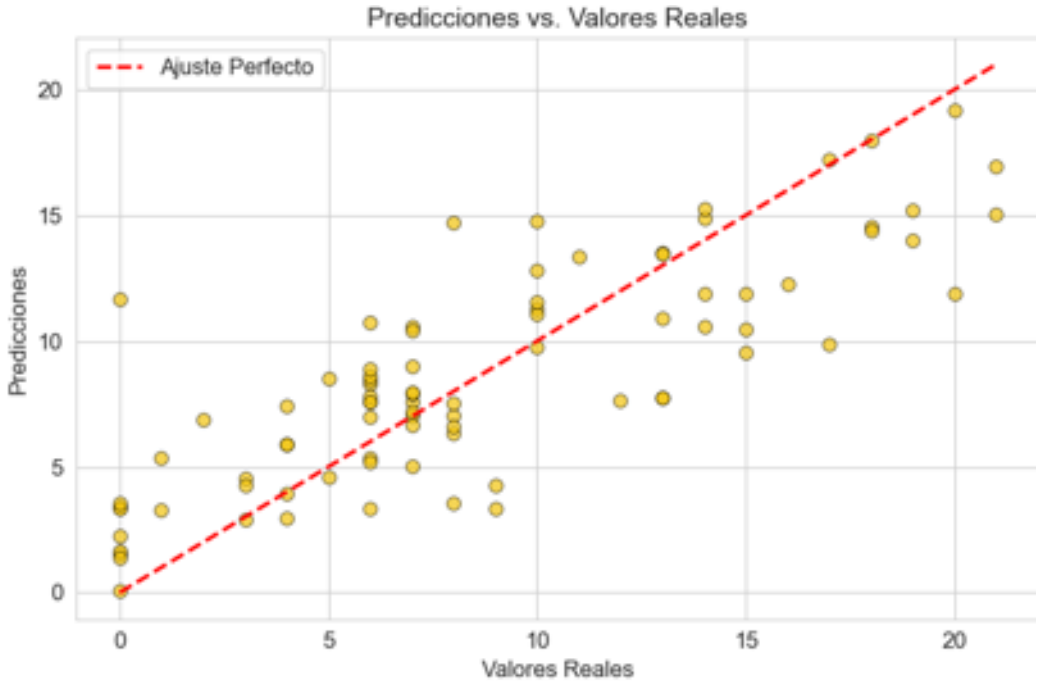


Ilustración 15 Predicciones vs. Valores Reales PHQ9 LightGBM Regressor

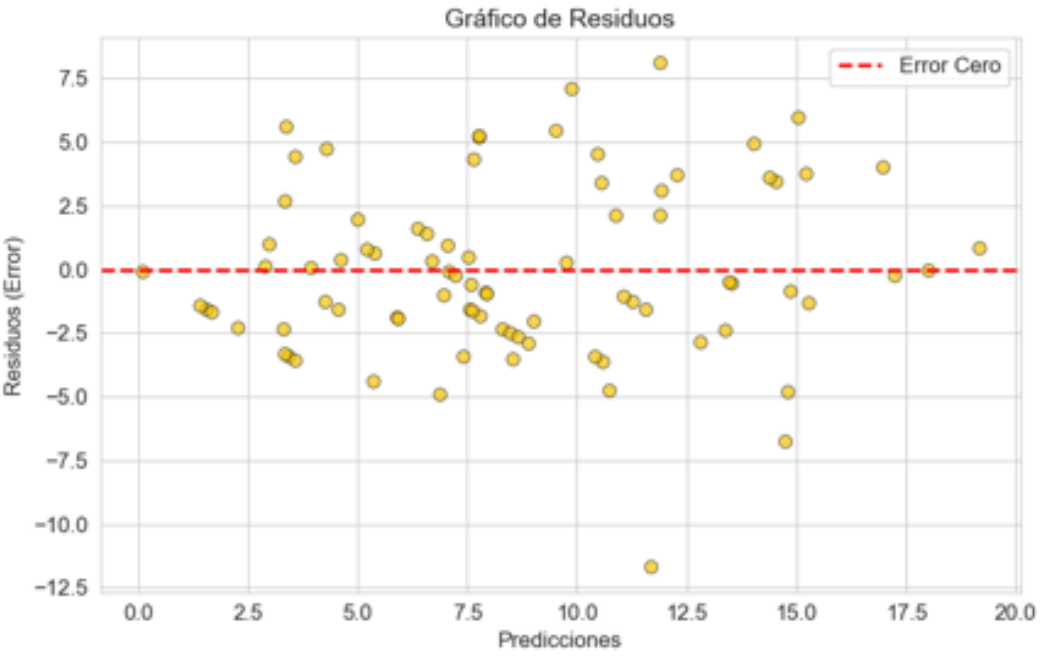


Ilustración 16 Gráfico de Residuos PHQ9 LightGBM Regressor

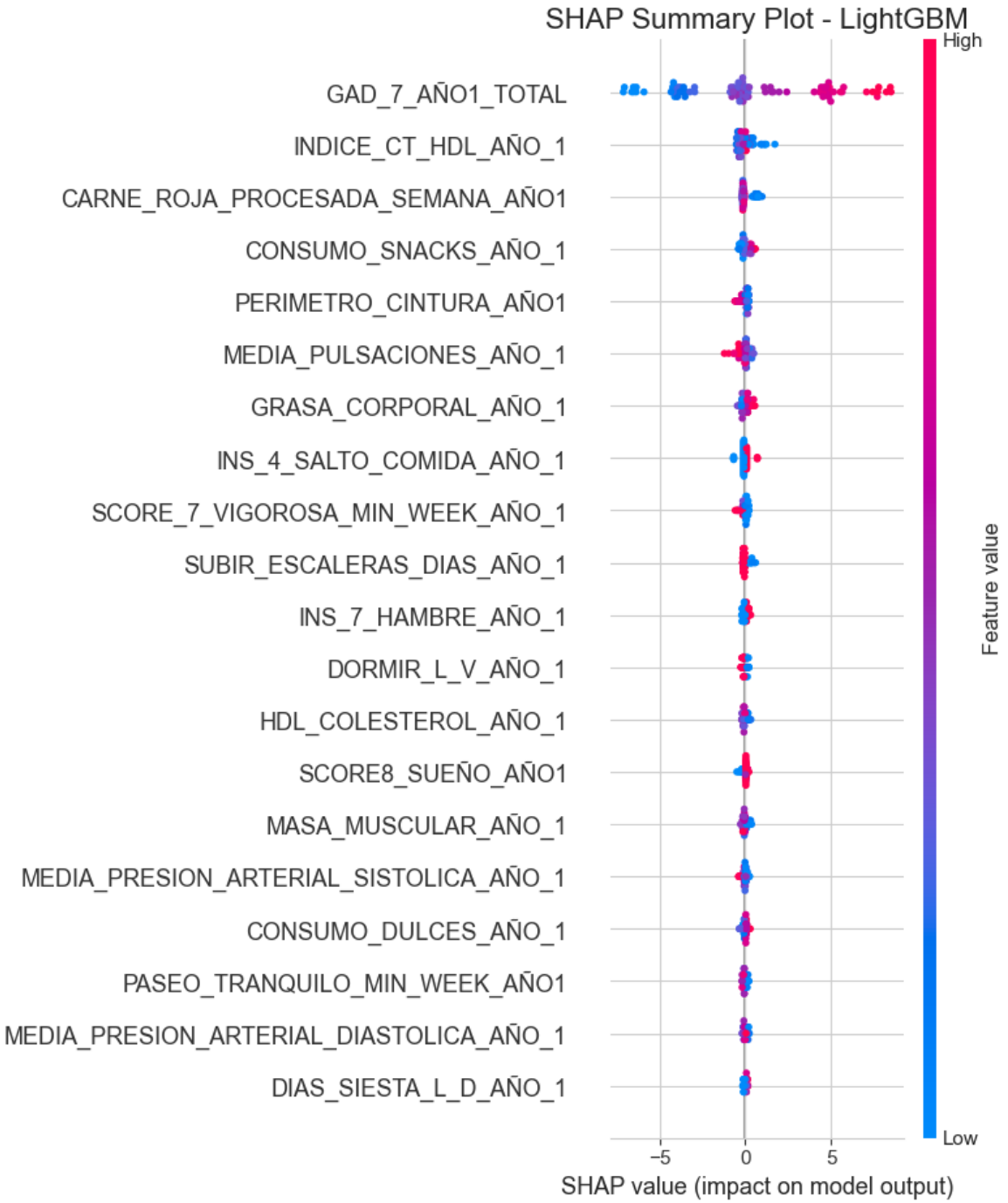


Ilustración 17 Importancia Variables PHQ9 LightGBM Regressor

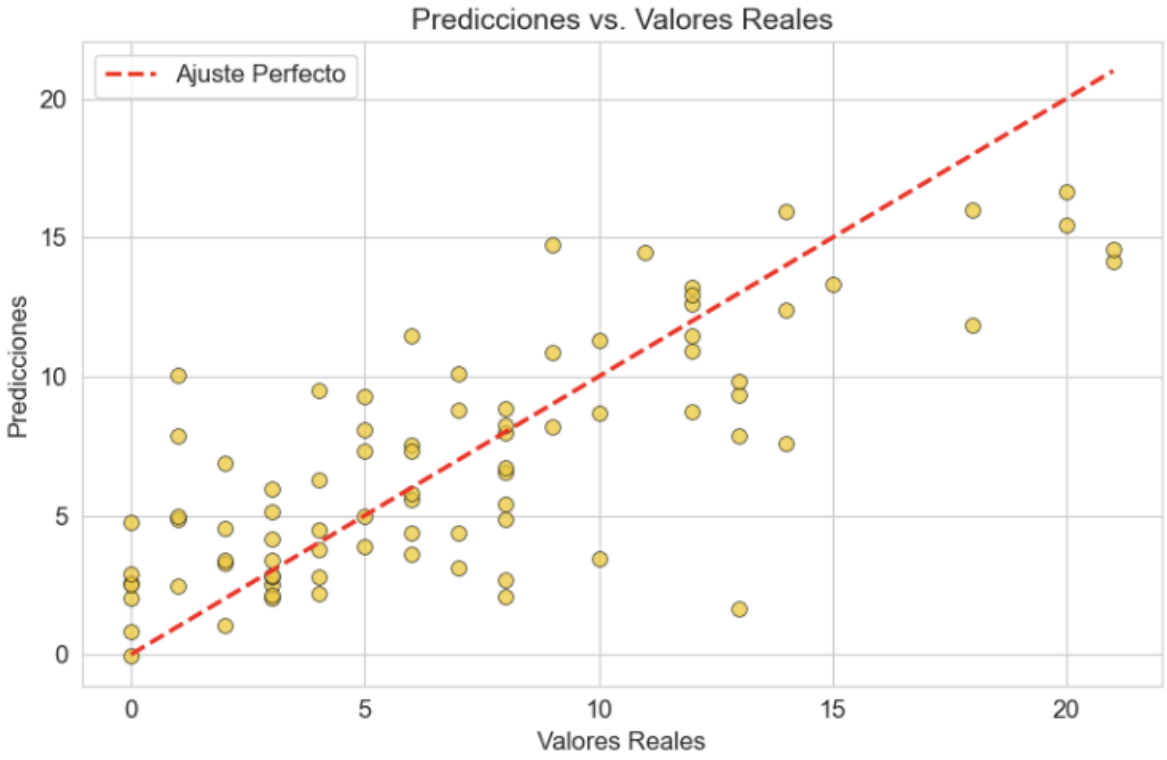


Ilustración 18 Predicciones vs. Valores Reales GAD7 LightGBM Regressor

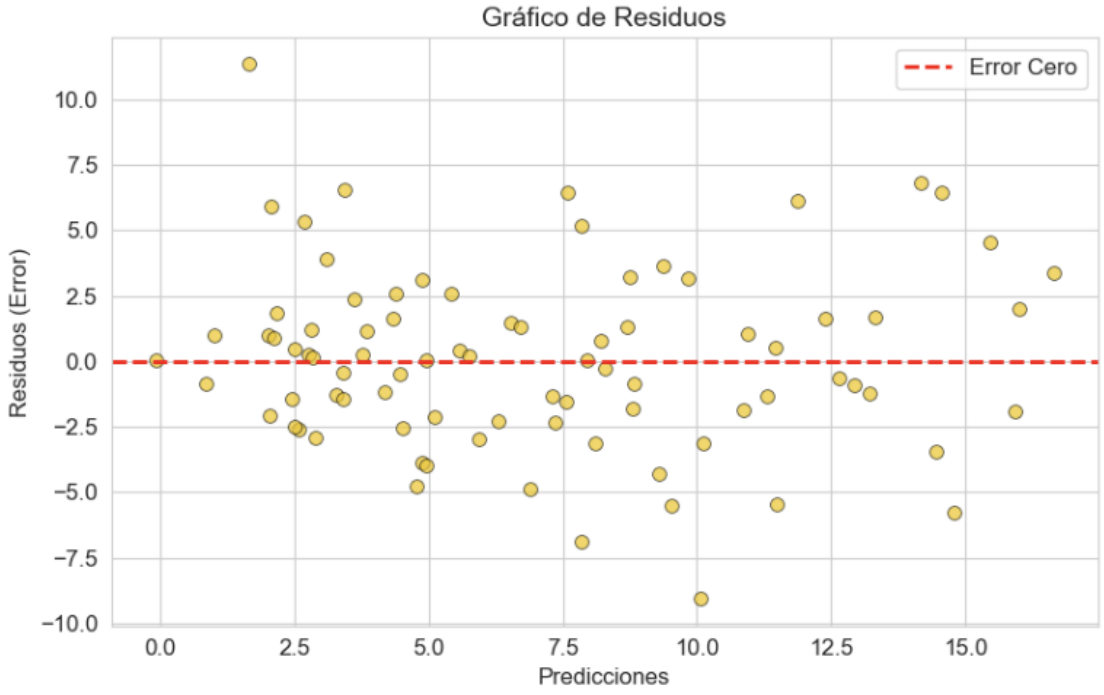


Ilustración 19 Gráfico de Residuos GAD7 LightGBM Regressor

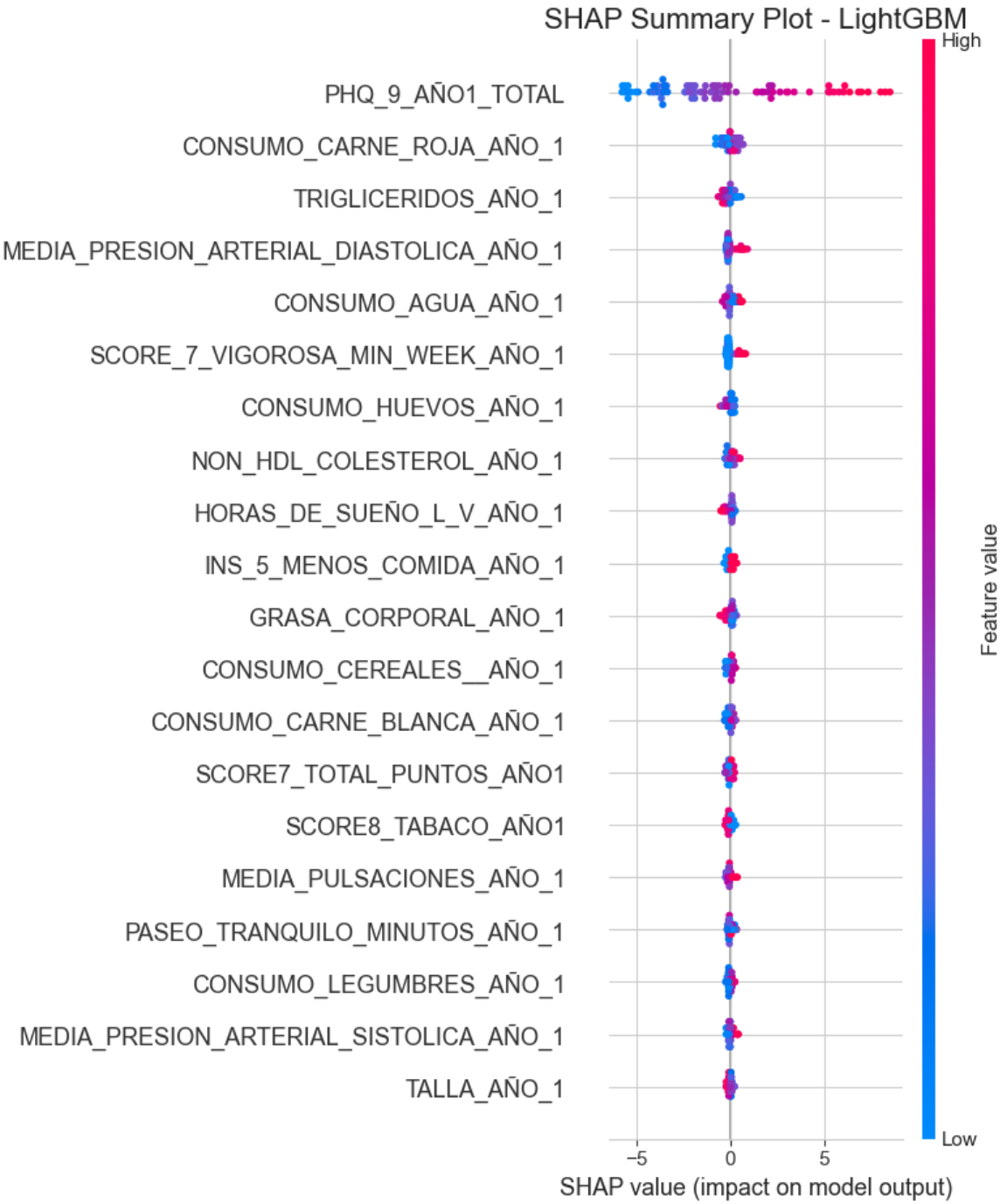


Ilustración 20 Importancia Variables GAD7 LightGBM Regressor

5.3.4 XGBOOST REGRESSOR

Para validar la eficacia de LightGBM, es imperativo compararlo con su principal competidor y referente de la industria, XGBoost (eXtreme Gradient Boosting). Su reputación se basa en una combinación de precisión, robustez y un diseño de sistema altamente optimizado.

Aunque comparte la misma filosofía de boosting que LightGBM, XGBoost presenta particularidades en su formulación matemática que lo distinguen. La más notable es el uso de una aproximación de segundo orden de Taylor para optimizar la función de pérdida. Esto significa que utiliza tanto el gradiente (primera derivada) como el hessiano (segunda derivada), proporcionando información más precisa sobre la curvatura de la función de pérdida y permitiendo una convergencia más rápida y exacta. Adicionalmente, XGBoost integra de forma nativa la regularización L1 (Lasso) y L2 (Ridge) en su función objetivo, penalizando no solo la magnitud de las puntuaciones de las hojas, sino también el número de hojas, lo que otorga un control exhaustivo sobre la complejidad del modelo para prevenir el sobreajuste.

Métricas en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.6113, MAE: 2.6624, RMSE: 3.5802

Métricas en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.4615, MAE: 2.8784, RMSE: 3.9327

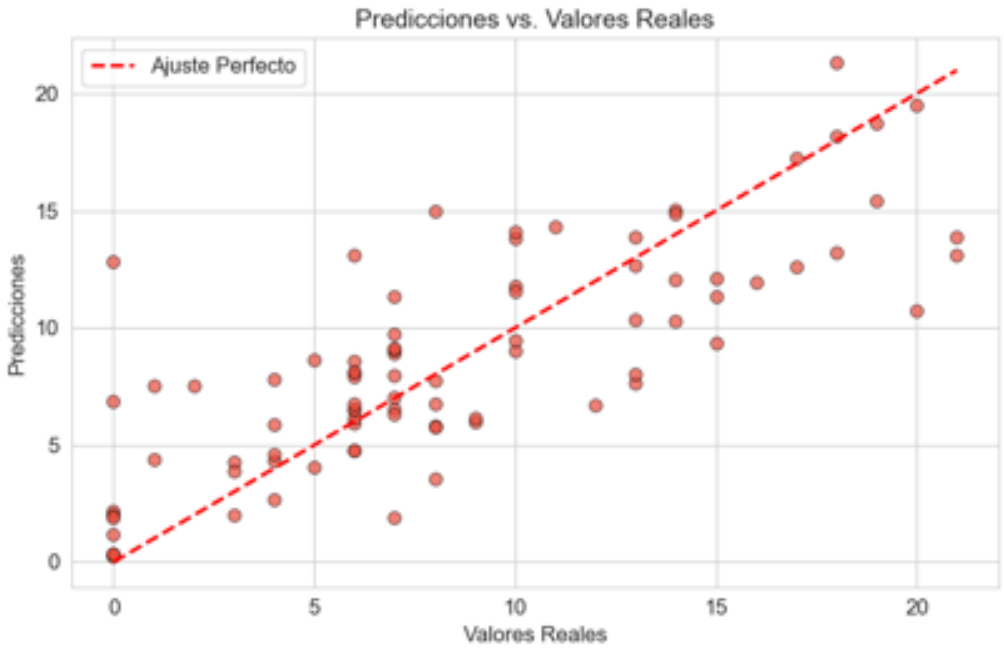


Ilustración 21 Predicciones vs. Valores Reales PHQ9 XGBoost Regressor

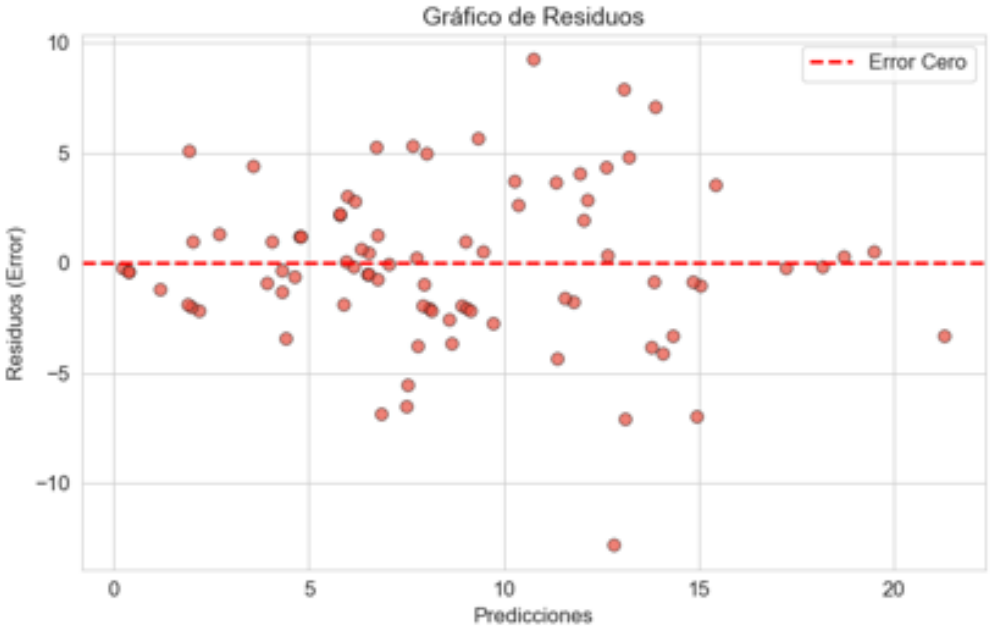


Ilustración 22 Gráfico de Residuos PHQ9 XGBoost Regressor

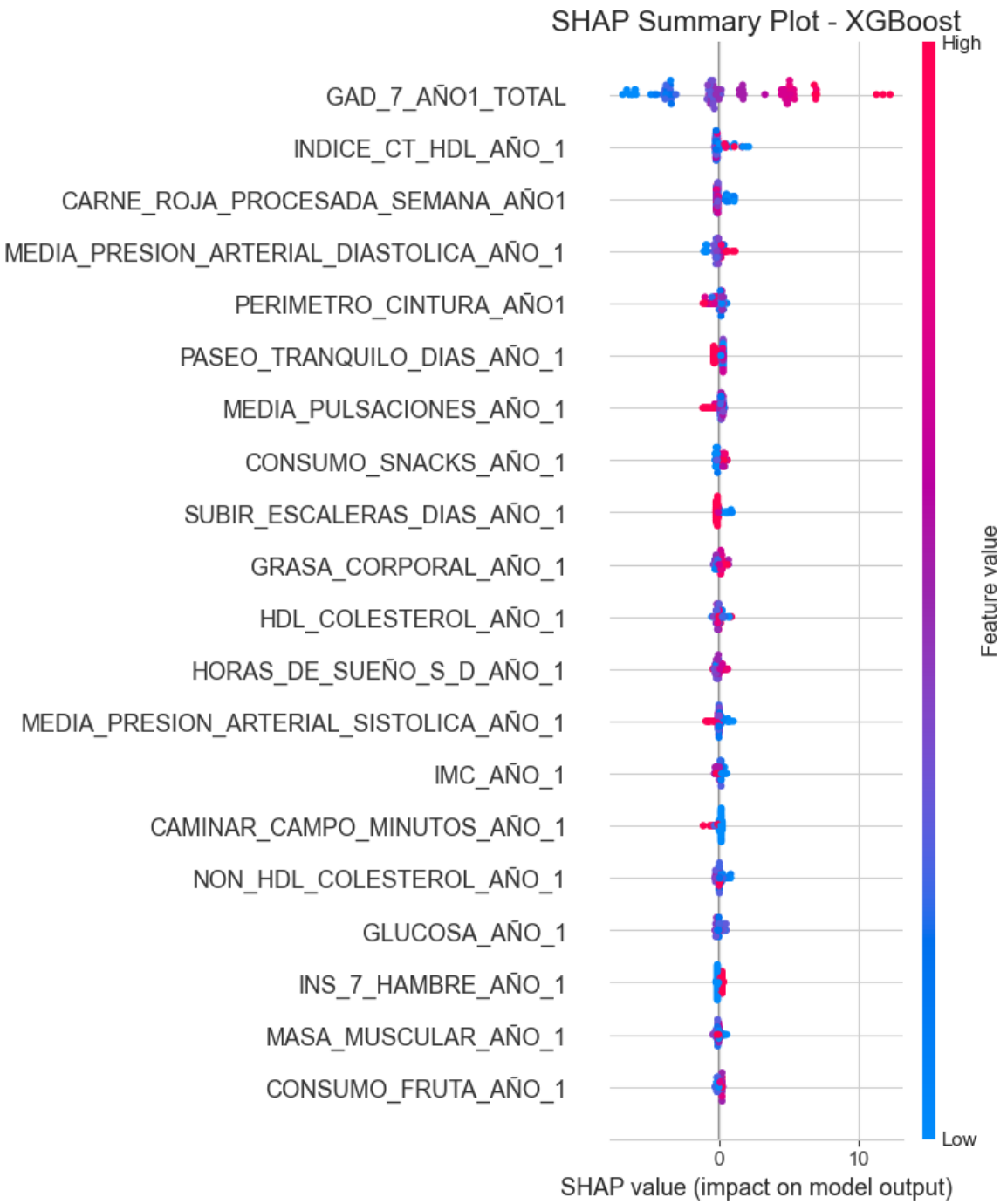


Ilustración 23 Importancia Variables PHQ9 XGBoost Regressor

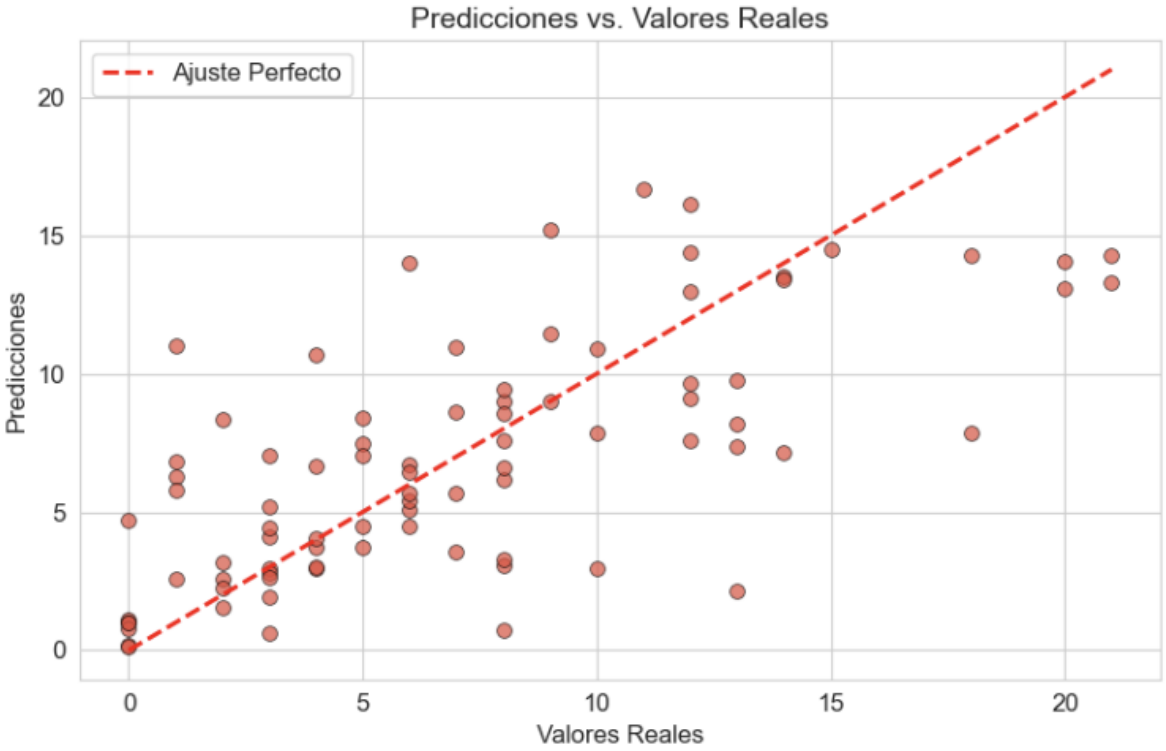


Ilustración 24 Predicciones vs. Valores Reales GAD7 XGBoost Regressor

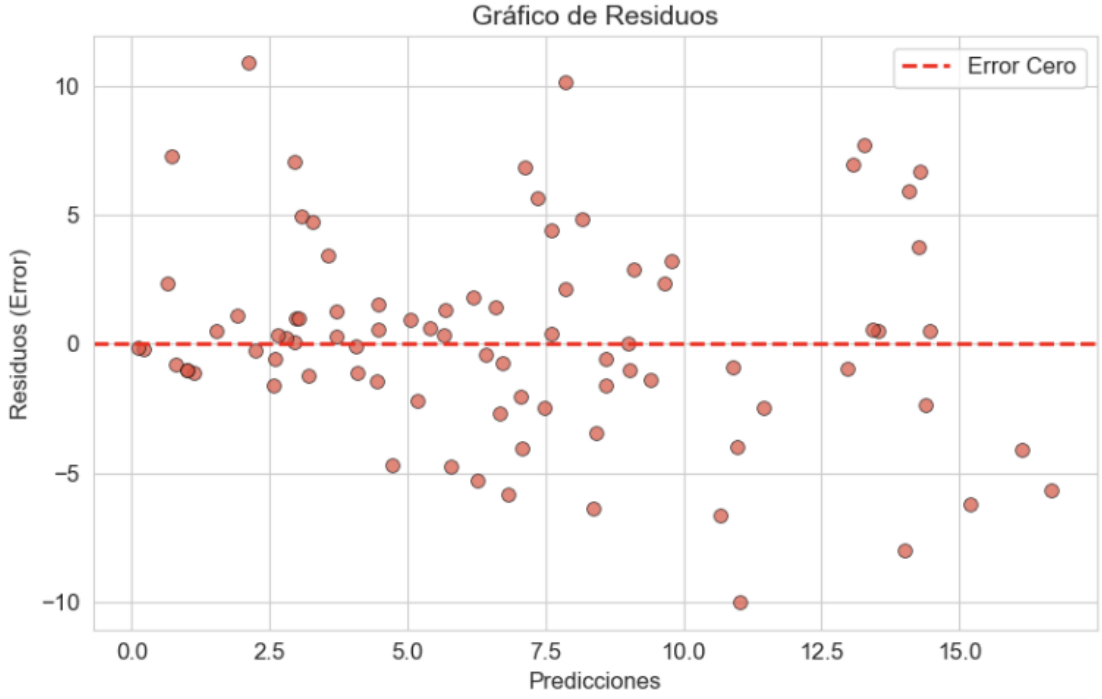


Ilustración 25 Gráfico de Residuos GAD7 XGBoost Regressor

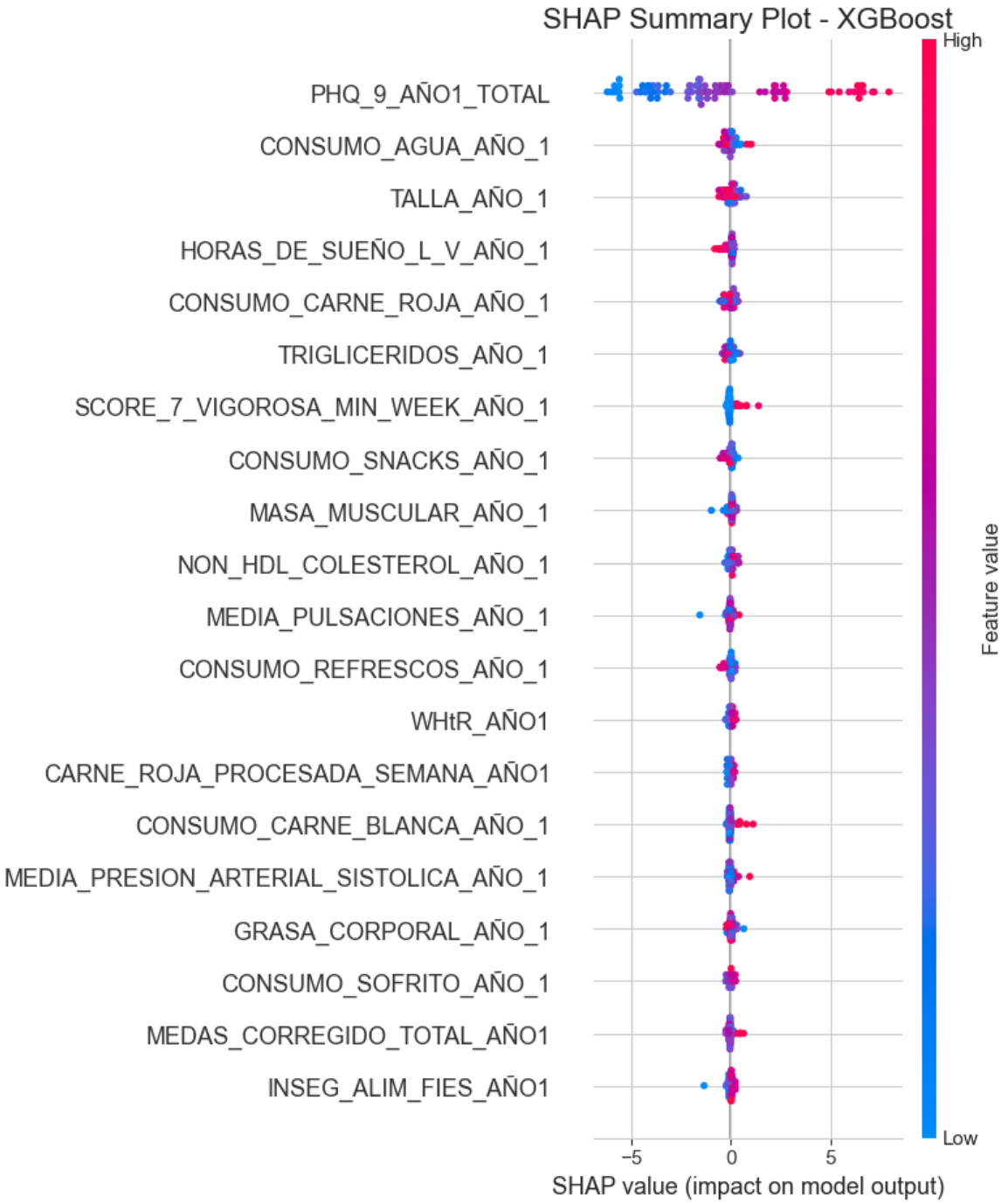


Ilustración 26 Importancia Variables GAD7 XGBoost Regressor

5.3.5 CATBOOST REGRESSOR

El análisis de los modelos de boosting concluye con CatBoost, cuya inclusión se justifica por su diseño innovador y especializado para el manejo de variables categóricas. Este es a menudo un punto débil en otros algoritmos, que requieren un preprocesamiento manual propenso a errores o a la pérdida de información.

La innovación central de CatBoost es el uso de Ordered Target Statistics. Para cada instancia de entrenamiento, el modelo codifica las variables categóricas calculando una estadística de la variable objetivo (por ejemplo, el promedio), pero lo hace utilizando únicamente las observaciones que aparecen antes en una permutación aleatoria del tiempo de los datos. Este mecanismo ordenado es su solución elegante y efectiva para prevenir el target leakage. Adicionalmente, CatBoost utiliza árboles de decisión simétricos (oblivious trees), donde todos los nodos de un mismo nivel utilizan la misma característica para dividirse. Esta estructura simétrica actúa como una forma de regularización implícita, previene el sobreajuste y permite una evaluación del modelo extremadamente rápida.

Métricas en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.5970, MAE: 2.8141, RMSE: 3.6451

Métricas en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.5617, MAE: 2.6390, RMSE: 3.5478

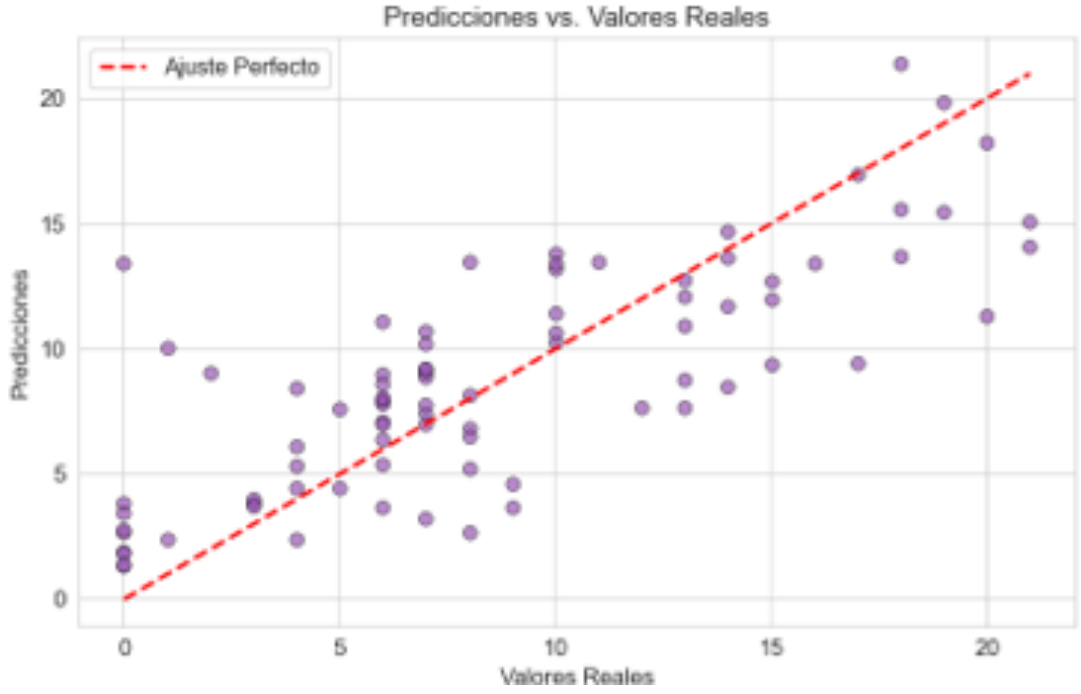


Ilustración 27 Predicciones vs. Valores Reales PHQ9 CatBoost Regressor

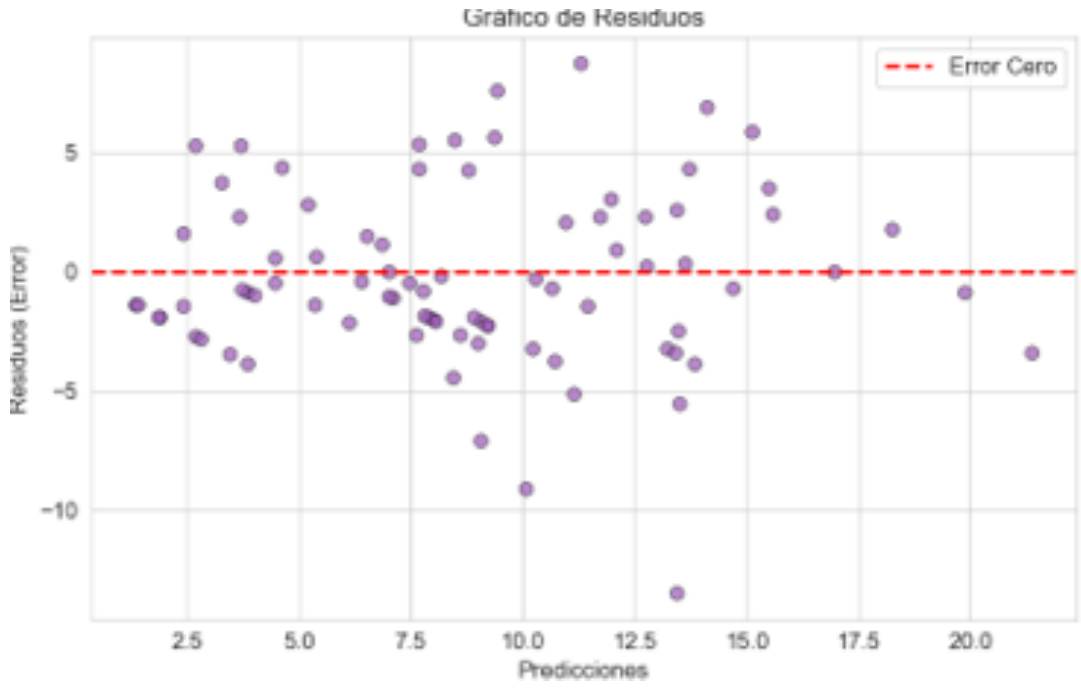


Ilustración 28 Gráfico de Residuos PHQ9 CatBoost Regressor

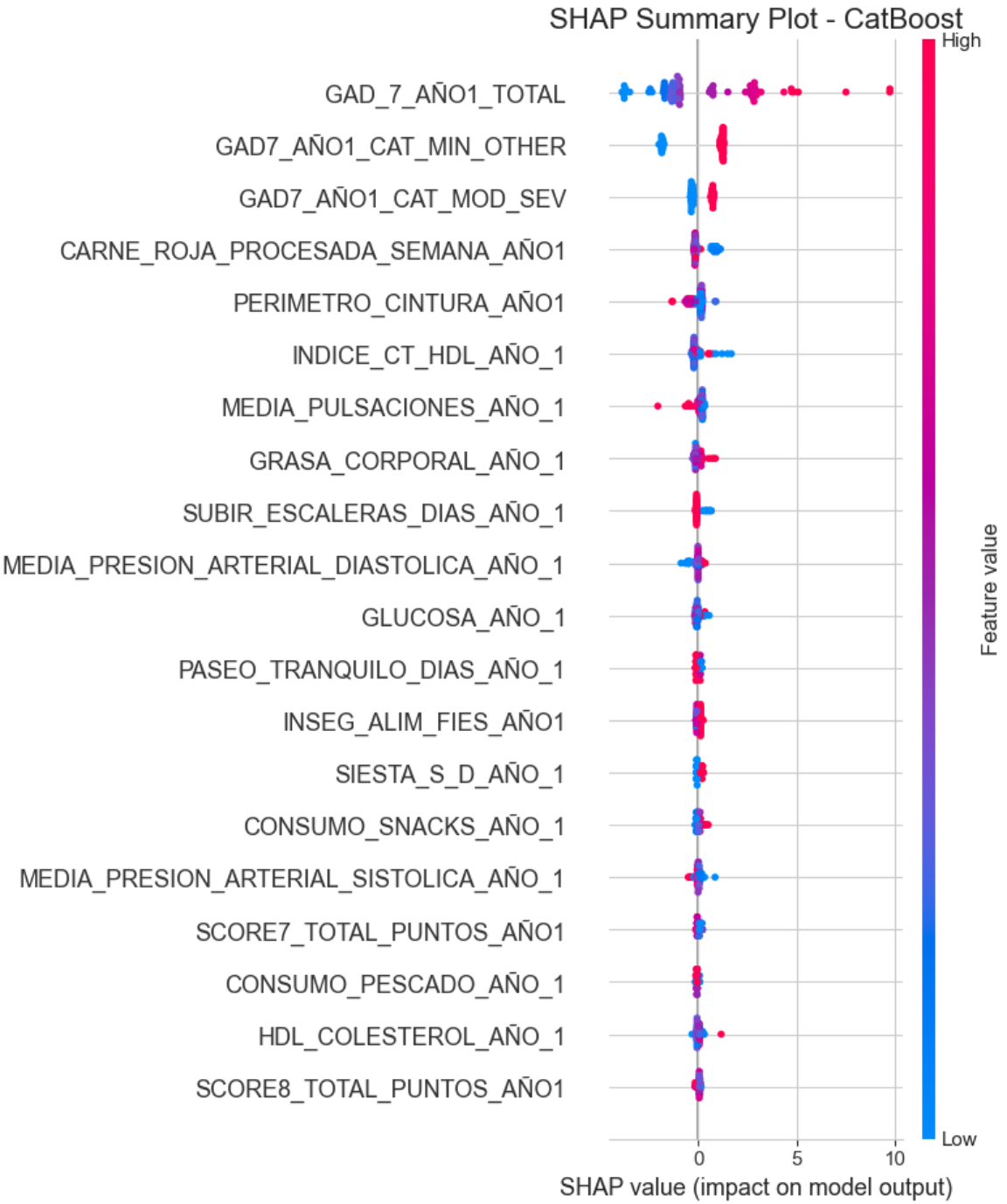


Ilustración 29 Importancia Variables PHQ9 CatBoost Regressor

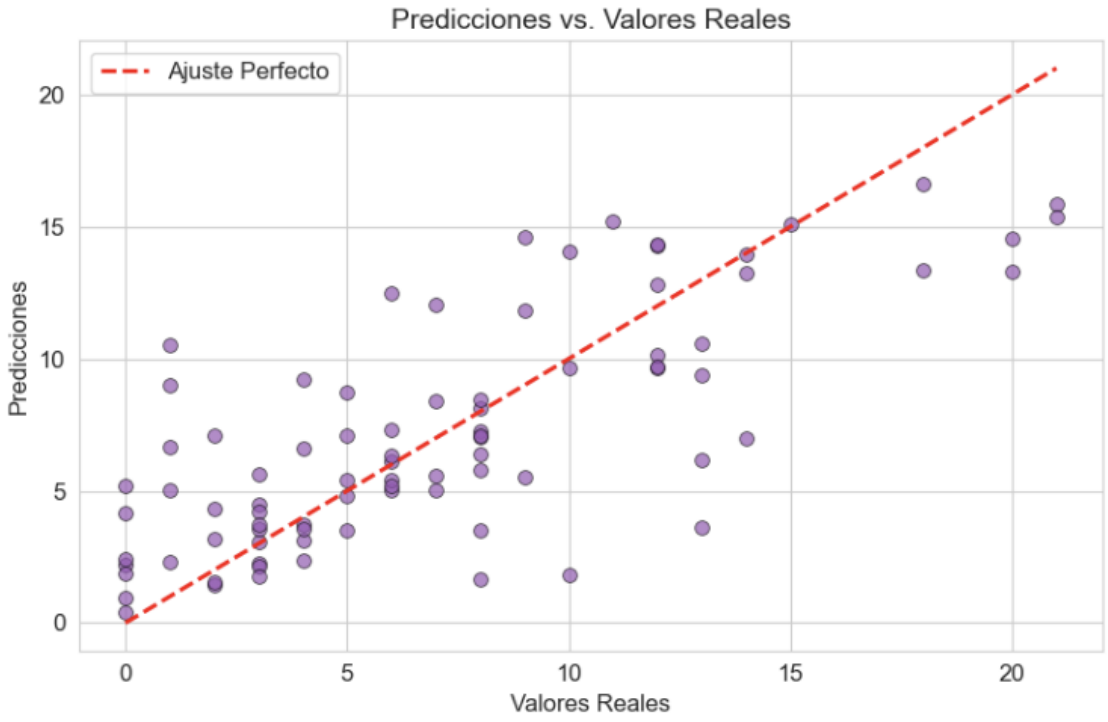


Ilustración 30 Predicciones vs. Valores Reales GAD7 CatBoost Regressor

Ilustración 31 Gráfico de Residuos GAD7 CatBoost Regressor

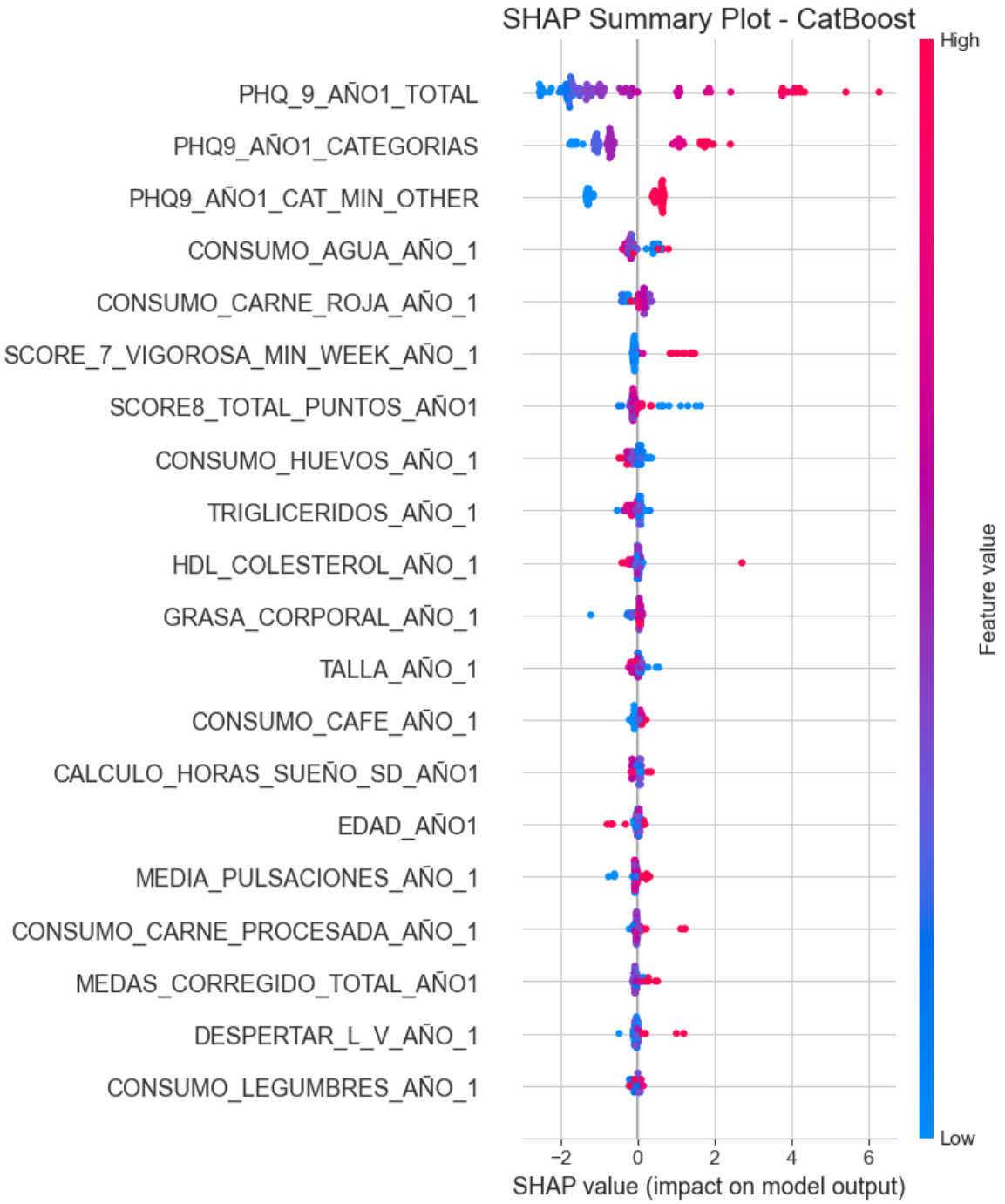


Ilustración 32 Importancia Variables GAD7 CatBoost Regressor

5.3.6 STACKING REGRESSOR

La fase culminante de la estrategia de modelado es la implementación de un Stacking Regressor, una técnica de meta-ensamblaje diseñada para combinar sinérgicamente las fortalezas de los modelos desarrollados previamente.

Su arquitectura de dos niveles es fundamental para su éxito. En el Nivel 0, los modelos más potentes y diversos (Random Forest, LightGBM, XGBoost y CatBoost) actúan como "aprendices base". Para generar las entradas del siguiente nivel sin fuga de información, sus predicciones se obtienen mediante un procedimiento de validación cruzada. El conjunto de entrenamiento se divide en K pliegues; para cada pliegue, los modelos base se entrenan en los K-1 pliegues restantes y hacen predicciones sobre el pliegue de validación. Al final, se obtiene un conjunto completo de predicciones "out-of-fold".

En el Nivel 1, estas predicciones se utilizan como el conjunto de características para entrenar un "meta-modelo". Para este rol, se ha seleccionado la Regresión de Ridge. La elección de un modelo lineal simple y regularizado es deliberada: las predicciones de los modelos base suelen estar altamente correlacionadas, y un modelo simple como Ridge es capaz de encontrar una combinación lineal óptima de estas predicciones sin sobreajustarse a ellas. El objetivo final es producir una predicción más precisa y robusta que la que cualquier modelo base podría lograr de forma individual.

Métricas de Stacking en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.6155, MAE: 2.7305, RMSE: 3.5609

Métricas de Stacking en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.5681, MAE: 2.5764, RMSE: 3.5220

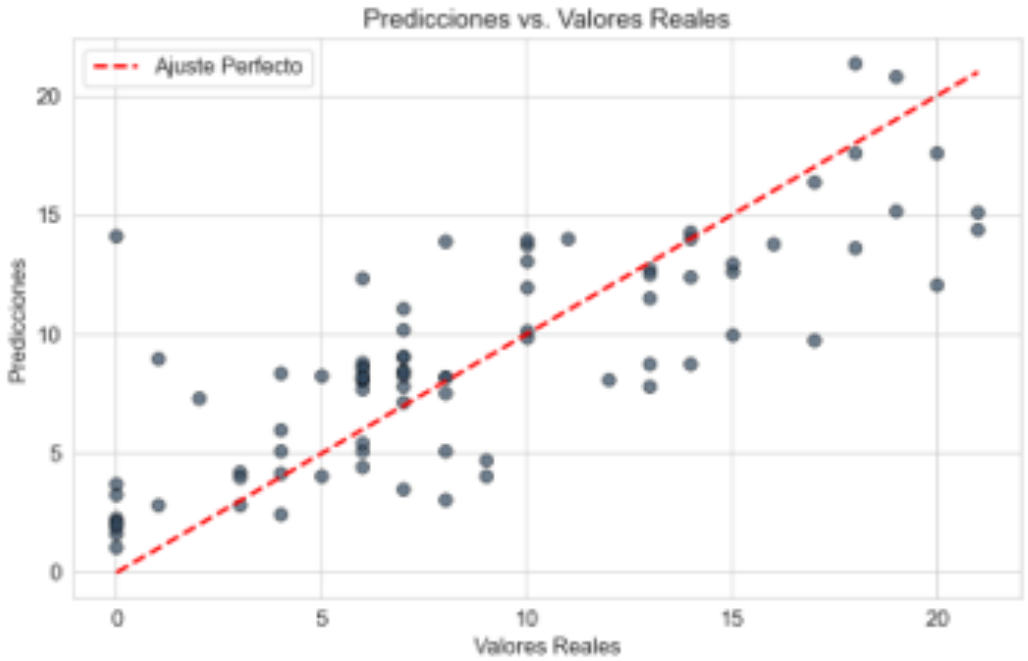


Ilustración 33 Predicciones vs. Valores Reales PHQ9 Stacking Regressor

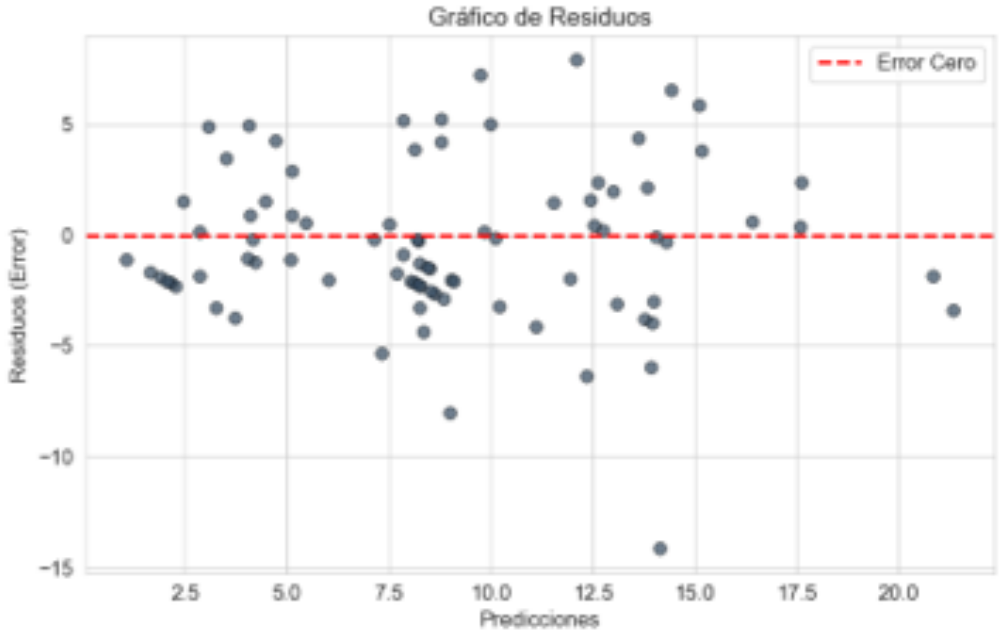


Ilustración 34 Gráfico de Residuos PHQ9 Stacking Regressor

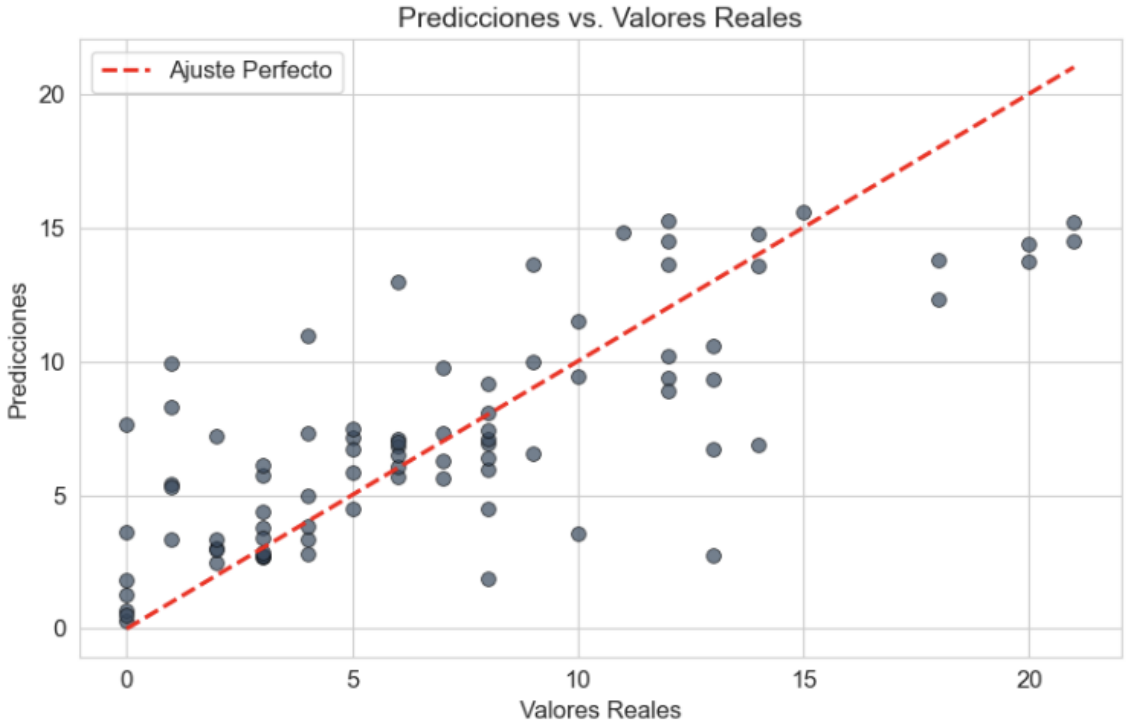


Ilustración 35 Predicciones vs. Valores Reales GAD7 Stacking Regressor

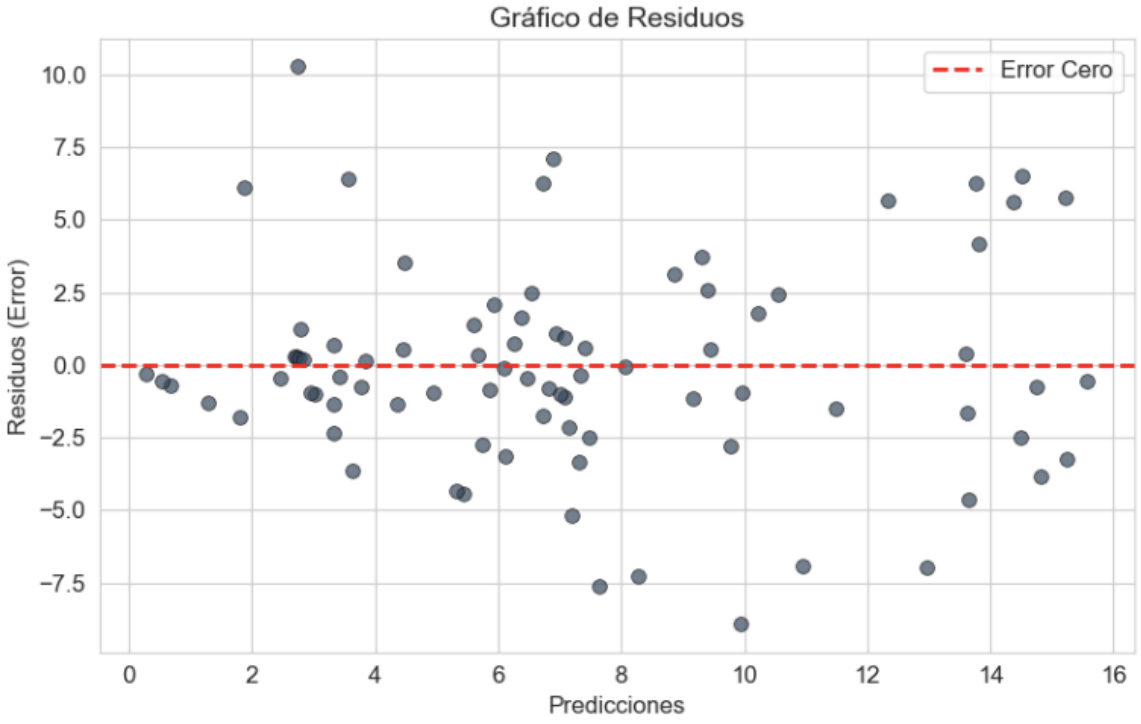


Ilustración 36 Gráfico de Residuos GAD7 Stacking Regressor

5.3.7 RIDGE FEDERADO (APRENDIZAJE FEDERADO MANUAL)

Una vez evaluado el rendimiento de diversos modelos en un entorno centralizado, donde se asume el acceso a la totalidad de los datos, el siguiente paso metodológico en este proyecto es explorar un paradigma de entrenamiento alternativo y más realista para contextos con datos sensibles: el Aprendizaje Federado (Federated Learning, FL). Para esta simulación, hemos seleccionado la Regresión de Ridge como el modelo base.

El enfoque tradicional y centralizado, utilizado en los modelos anteriores, requiere que todos los datos se recopilen en un único servidor para entrenar el modelo. Sin embargo, en numerosos dominios, como el de la salud, la psicología clínica o las finanzas, compartir datos brutos es inviable debido a estrictas regulaciones de privacidad y confidencialidad además de los motivos éticos.

El Aprendizaje Federado resuelve este problema invirtiendo el paradigma: en lugar de llevar los datos al modelo, el modelo se lleva a los datos. El entrenamiento se distribuye entre múltiples "clientes" (por ejemplo, hospitales, dispositivos móviles, etc.), y los datos brutos nunca abandonan su ubicación original. Únicamente se comparten los parámetros aprendidos por el modelo, no la información personal.

Nuestra simulación de Ridge Federado emula este proceso y se puede desglosar en las siguientes etapas iterativas, conocidas como "rondas de comunicación":

- Inicialización Central: El proceso comienza en un servidor central que inicializa un modelo de Regresión de Ridge "global". En esencia, esto significa crear un vector de coeficientes (β) y un intercepto, generalmente inicializados en cero. Este modelo global inicial se distribuye a todos los clientes participantes en la simulación.

- Entrenamiento Local y Privado: Cada cliente recibe una copia idéntica del modelo global. A continuación, entrena este modelo utilizando únicamente su propio conjunto de datos local. El objetivo de este entrenamiento es ajustar los

coeficientes del modelo Ridge para que se adapten lo mejor posible a los patrones presentes en sus datos locales, respetando la penalización alpha de Ridge.

- Comunicación de Parámetros: Una vez finalizado el entrenamiento local, cada cliente ha obtenido un conjunto de coeficientes actualizado. En este punto crucial, el cliente no devuelve sus datos privados. En su lugar, envía únicamente los parámetros de su modelo (el vector de coeficientes y el intercepto) de vuelta al servidor central. Esta es la esencia de la preservación de la privacidad.

- Agregación en el Servidor: El servidor central recibe los parámetros actualizados de todos los clientes. Su tarea es agregar esta información para crear un nuevo modelo global mejorado. El método más común, y el utilizado en nuestra simulación, es el Promediado Federado (Federated Averaging). El servidor calcula la media ponderada de todos los coeficientes recibidos, donde la ponderación suele ser proporcional al número de muestras que cada cliente utilizó en su entrenamiento.

$$\beta^{t+1}_{global} = \sum^{N}_{k=1} \beta^{t}_{local,\,k} \; n_k/n$$

Donde N es el número de clientes, n_k es el número de muestras del cliente k, y n es el número total de muestras.

- Iteración y Convergencia: El conjunto de coeficientes promediado se convierte en el nuevo modelo global para la siguiente ronda. Este modelo, que ahora contiene el conocimiento colectivo de todos los clientes, se distribuye de nuevo, y el proceso (pasos 2-4) se repite. Tras varias rondas de comunicación, el modelo global converge hacia una solución robusta que generaliza bien a través de todos los datos distribuidos, sin que el servidor central haya accedido nunca a una sola muestra de datos privados.

La elección de la Regresión de Ridge no es casual. Su naturaleza lineal lo convierte en el candidato perfecto para demostrar el concepto de Aprendizaje Federado de manera clara y efectiva. Los "parámetros" del modelo son simplemente un vector de coeficientes y un intercepto. El proceso de promediar estos vectores es matemáticamente sencillo, computacionalmente eficiente e intuitivo. Intentar federar modelos más complejos, como un Random Forest, donde "promediar" estructuras de árboles no es una operación trivial, sería considerablemente más complejo.

En conclusión, este experimento con Ridge Federado nos permite evaluar la viabilidad y el rendimiento de un enfoque que prioriza la privacidad, comparando su precisión con la de los modelos centralizados para cuantificar el posible "coste" en rendimiento que supone no tener acceso directo a todos los datos.

Métricas finales en Test para predecir PHQ_9_AÑO1_TOTAL: R^2: 0.4691, MAE: 3.5604,

RMSE: 4.1840

Métricas finales en Test para predecir GAD_7_AÑO1_TOTAL: R^2: 0.4129, MAE: 3.1196,

RMSE: 4.1061

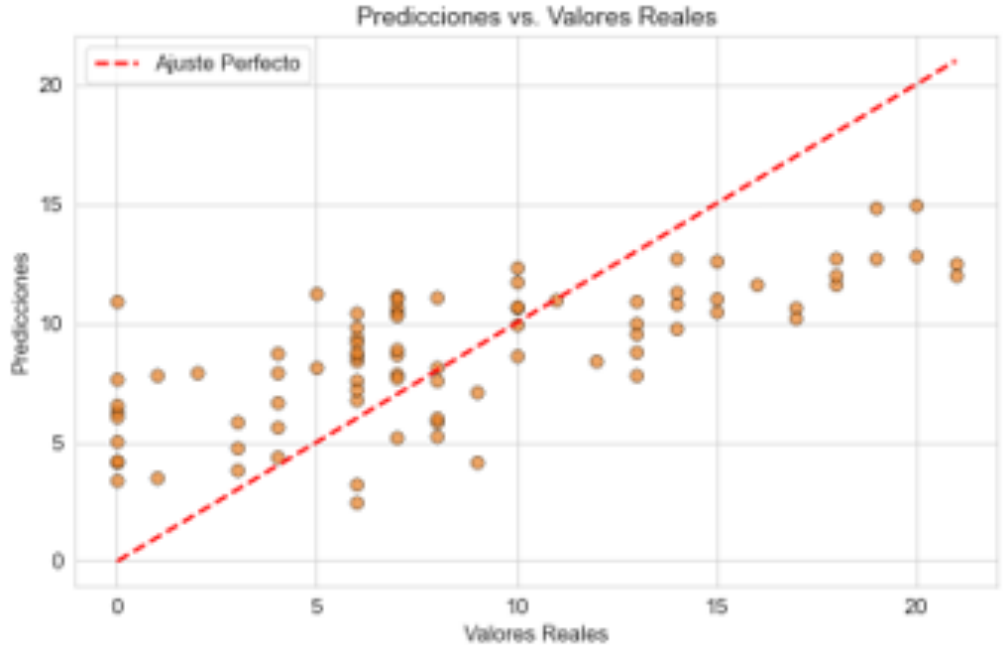


Ilustración 37 Predicciones vs. Valores Reales PHQ9 modelo Ridge Federado

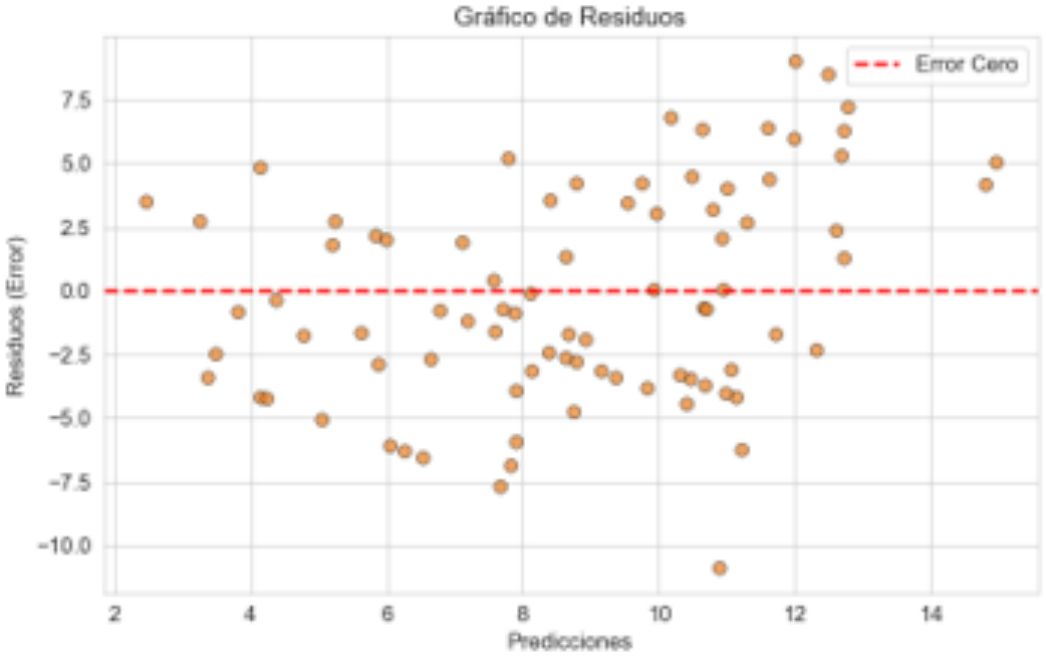


Ilustración 38 Gráfico de Residuos PHQ9 Modelo Ridge Federado

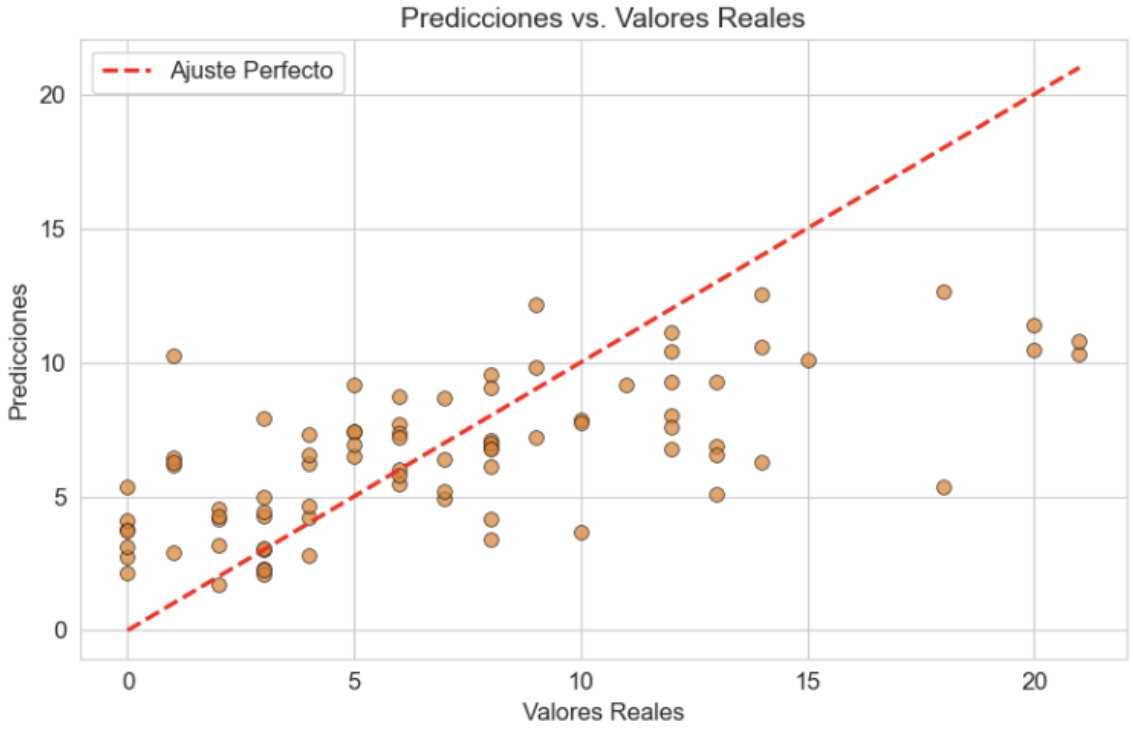


Ilustración 39 Predicciones vs. Valores Reales GAD7 modelo Ridge Federado

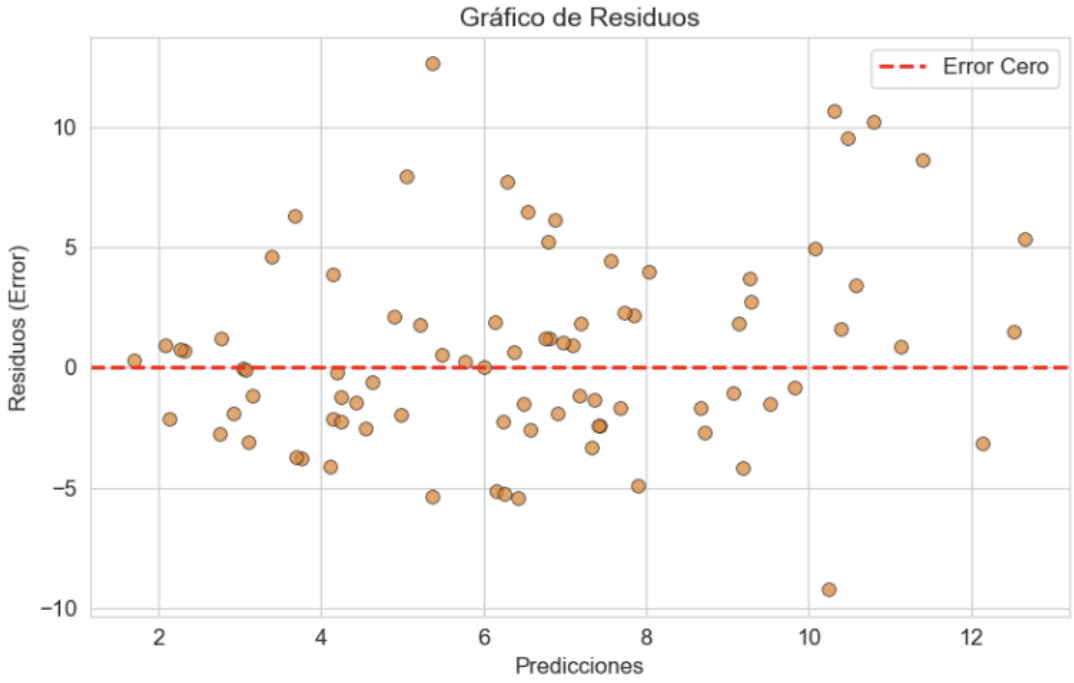


Ilustración 40 Gráfico de Residuos GAD7 Modelo Ridge Federado

5.4 DESARROLLO DE LOS MODELOS DE CLASIFICACIÓN: PREDICCIÓN DE CATEGORÍAS DISCRETAS

Tras haber abordado el problema desde una perspectiva de regresión para predecir valores continuos, el enfoque del proyecto se amplía ahora para abordar tareas de clasificación. El objetivo en esta fase es predecir la categoría o clase discreta a la que pertenece cada observación.

La decisión de analizar las variables objetivo también de forma categórica, y no solo continua, responde a una necesidad fundamental de alinear los resultados del modelo con la práctica clínica y la toma de decisiones del mundo real. En contextos de salud, una puntuación numérica (como la obtenida en un cuestionario) rara vez se utiliza en su forma cruda. En su lugar, se interpreta a través de umbrales diagnósticos que la convierten en una categoría clínicamente significativa (p. ej., depresión "leve", "moderada" o "severa"). Estas categorías son las que determinan el tipo de intervención, el seguimiento y el plan de tratamiento para un paciente.

5.4.1 LOGISTIC REGRESSION

La Regresión Logística, a pesar de su nombre, es el modelo fundamental para tareas de clasificación lineal. Se establece como nuestro punto de partida y base de referencia (baseline), cumpliendo el mismo rol que la Regresión de Ridge en el análisis de regresión. Su función es modelar la probabilidad de que una instancia pertenezca a una clase determinada.

El mecanismo central de la Regresión Logística consiste en aplicar una función logística (o sigmoide) al resultado de una combinación lineal de las características de entrada. Esta función mapea cualquier valor real a un rango entre 0 y 1, que se interpreta como la probabilidad de la clase positiva. Para problemas multiclase, este concepto se extiende mediante estrategias como One-vs-Rest (OvR) o, más comúnmente, a través de la función Softmax, que calcula una distribución de probabilidad sobre todas las clases posibles. La clase con la mayor probabilidad es la que se asigna como predicción final.

Su inclusión es crucial no solo por ser un benchmark interpretable y computacionalmente eficiente, sino también por su rol estratégico posterior como meta-modelo en el Stacking Classifier y como el modelo base para la simulación de aprendizaje federado en un contexto de clasificación.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.3929, F1-Score (Weighted): 0.387

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2976, F1-Score (Weighted): 0.2904

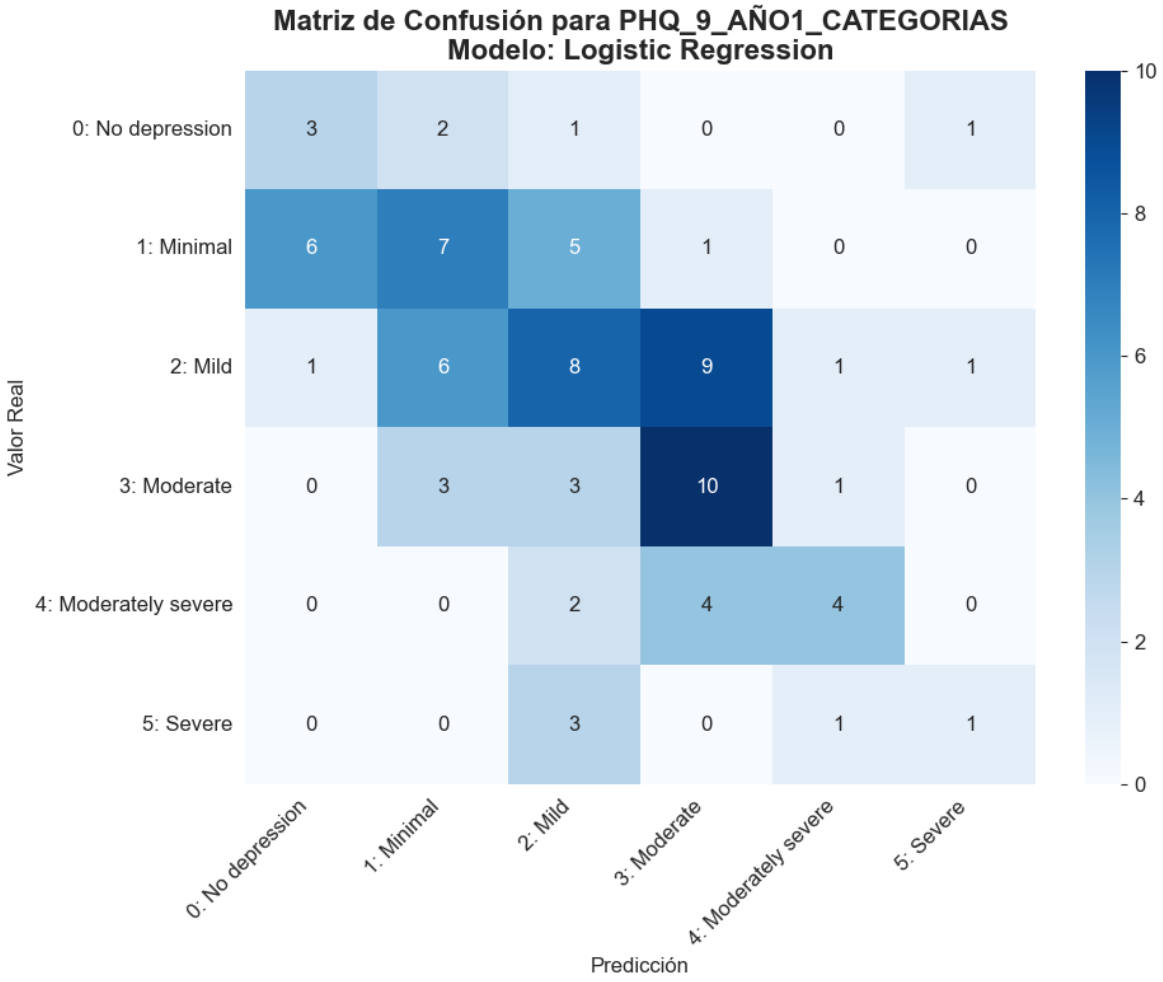


Ilustración 41 Matriz de Confusión PHQ9 Logistic Regression

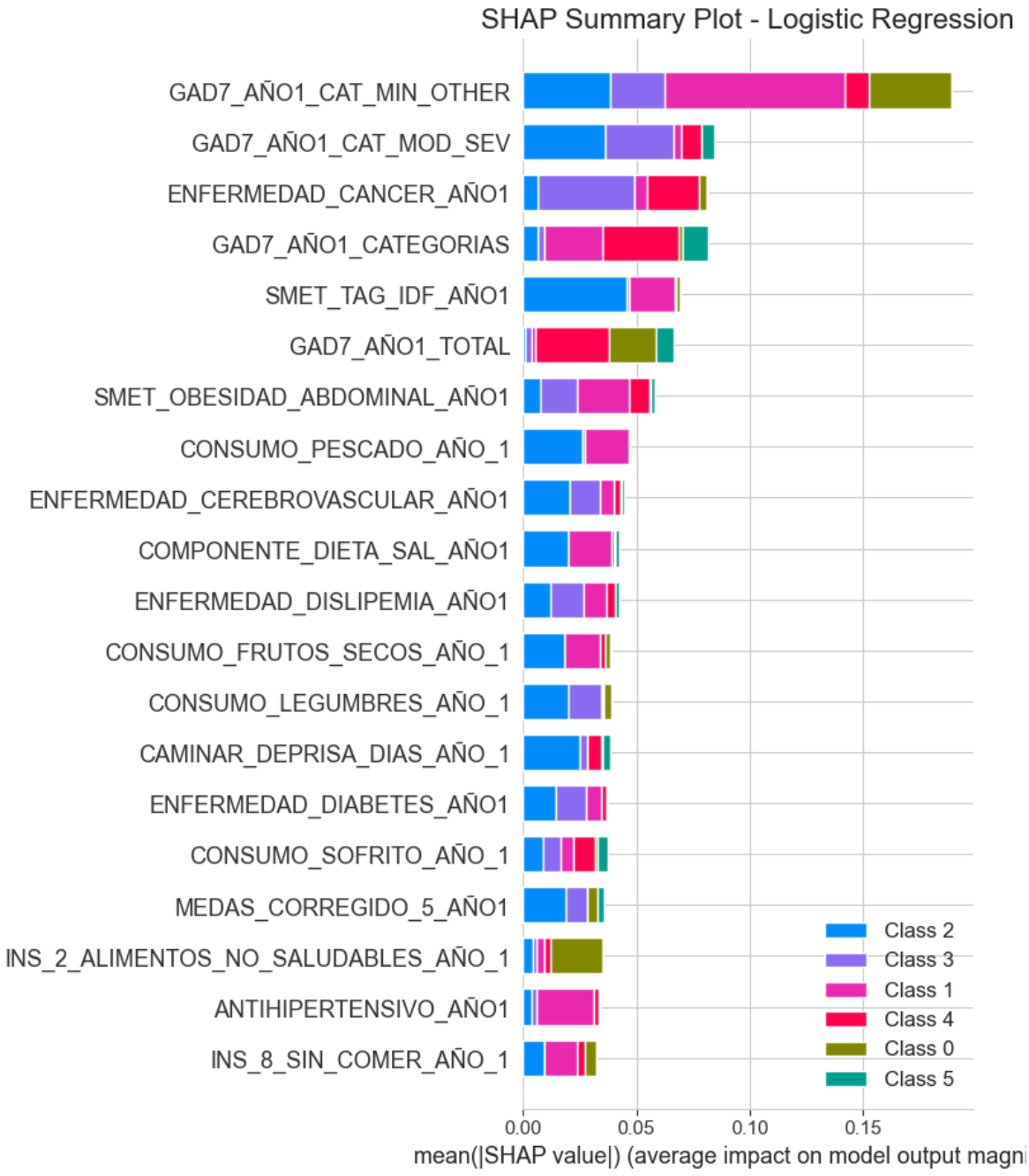


Ilustración 42 Importancia de variables PHQ9 Logistic Regression

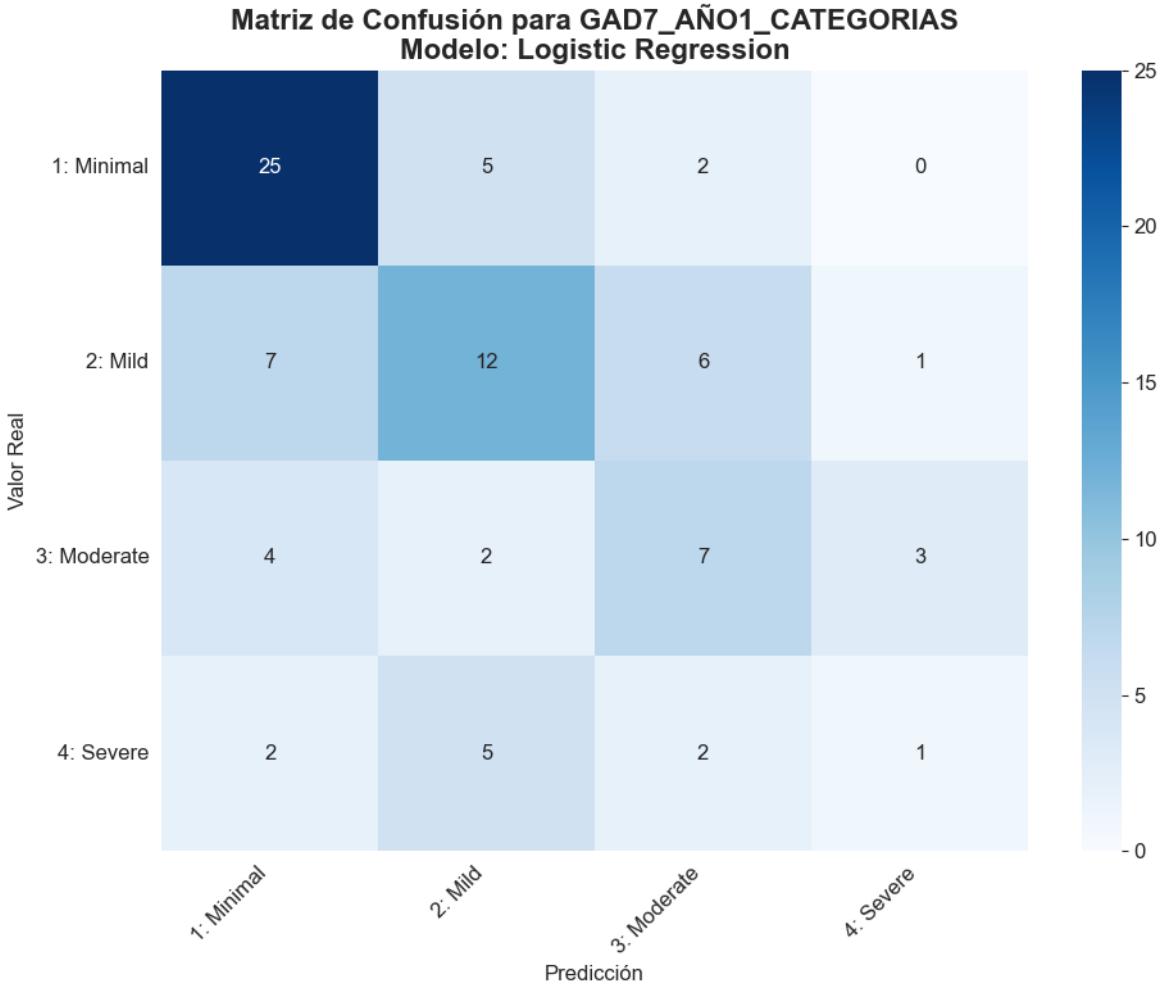


Ilustración 43 Matriz de Confusión GAD7 Logistic Regression

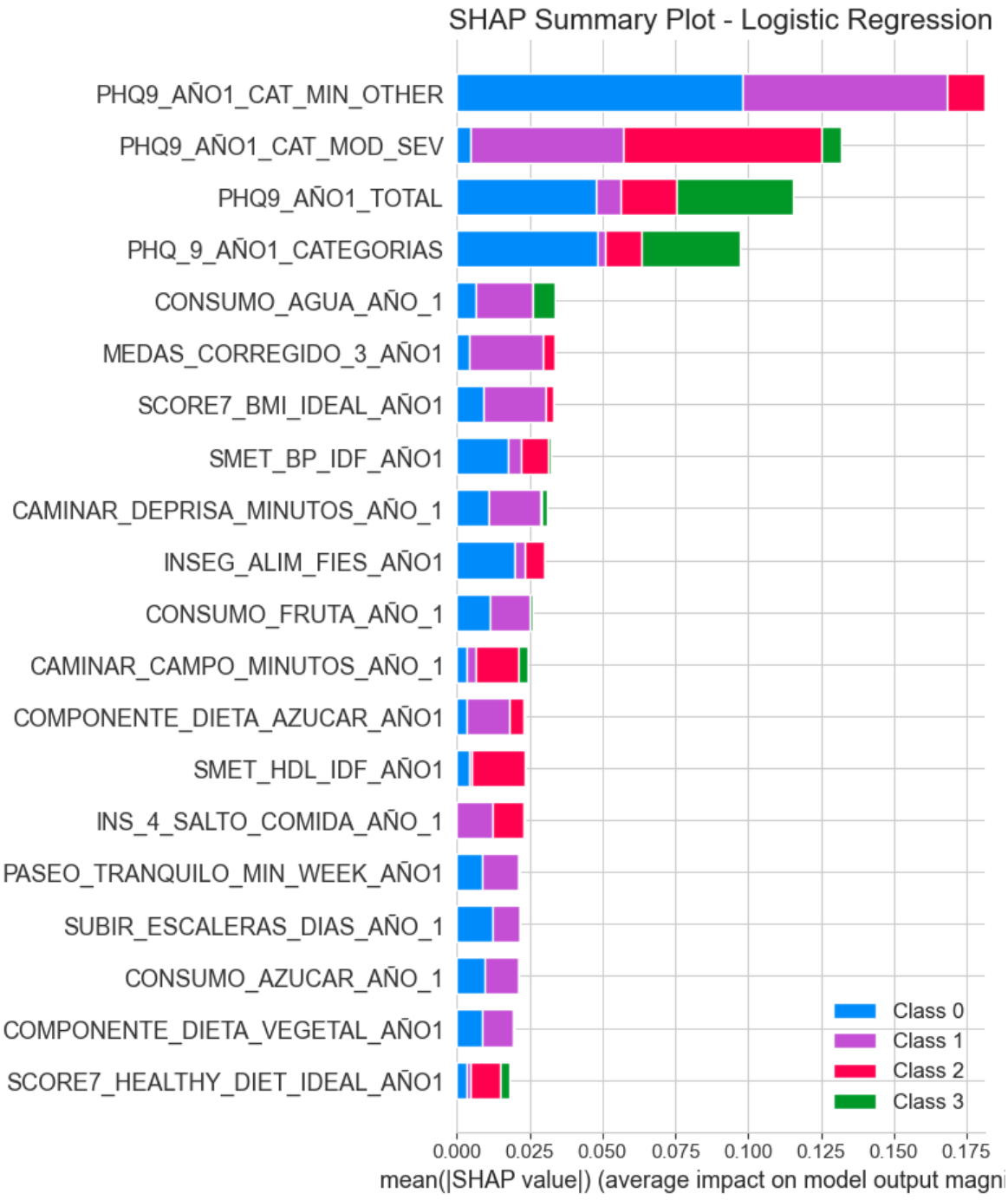


Ilustración 44 Importancia de variables GAD7 Logistic Regression

5.4.2 RANDOM FOREST CLASSIFIER

Una vez establecido el rendimiento del modelo lineal, se introduce el Random Forest Classifier para investigar si el problema de clasificación se beneficia de la captura de fronteras de decisión no lineales. Su estructura y funcionamiento son análogos a los de su contraparte de regresión, basándose en los principios de bagging y aleatoriedad de características para construir un ensamblaje de árboles de decisión decorrelacionados.

La diferencia fundamental reside en el mecanismo de agregación final. En lugar de promediar las predicciones, el Random Forest Classifier utiliza un sistema de votación por mayoría (majority voting). Cada uno de los cientos de árboles del bosque "vota" por una clase, y la clase que recibe la mayor cantidad de votos se convierte en la predicción final del ensamblaje. Este proceso de agregación democrática permite al modelo definir regiones de clasificación complejas y no lineales, reduciendo drásticamente la varianza y el riesgo de sobreajuste en comparación con un único árbol de decisión.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.4524, F1-Score (Weighted): 0.4331

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2738, F1-Score (Weighted): 0.2555

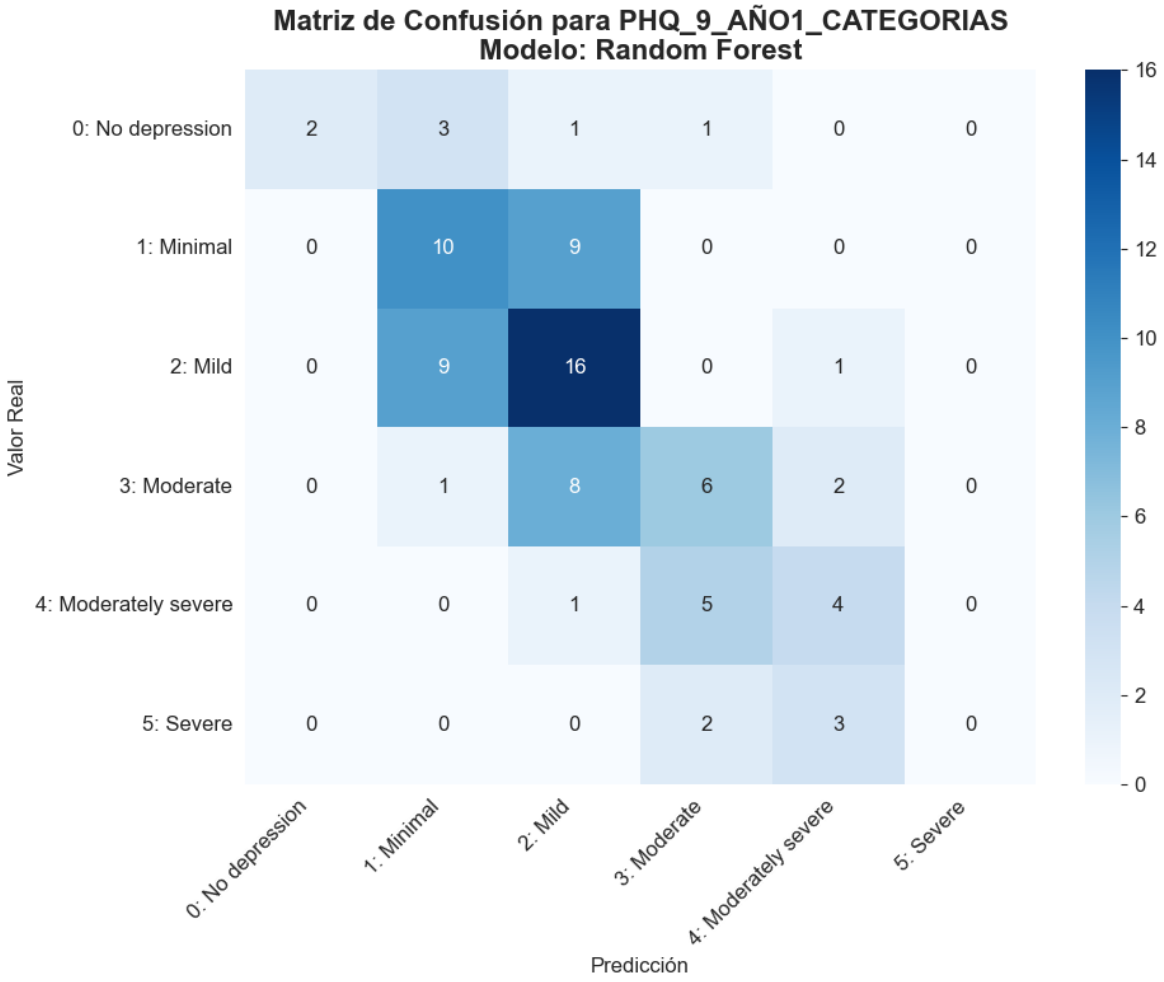


Ilustración 45 Matriz de Confusión PHQ9 RF Classifier

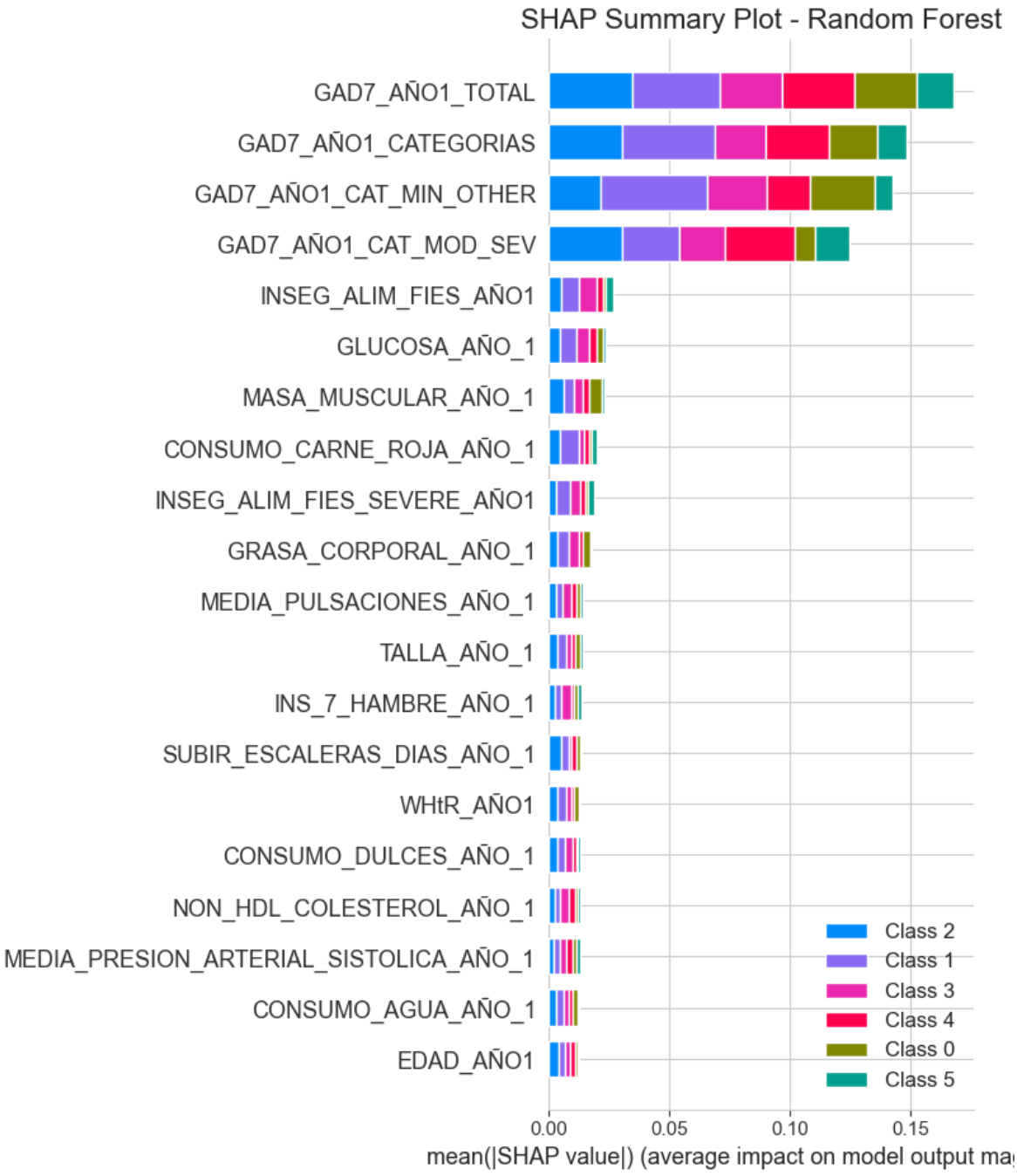


Ilustración 46 Importancia de variables PHQ9 RF Classifier

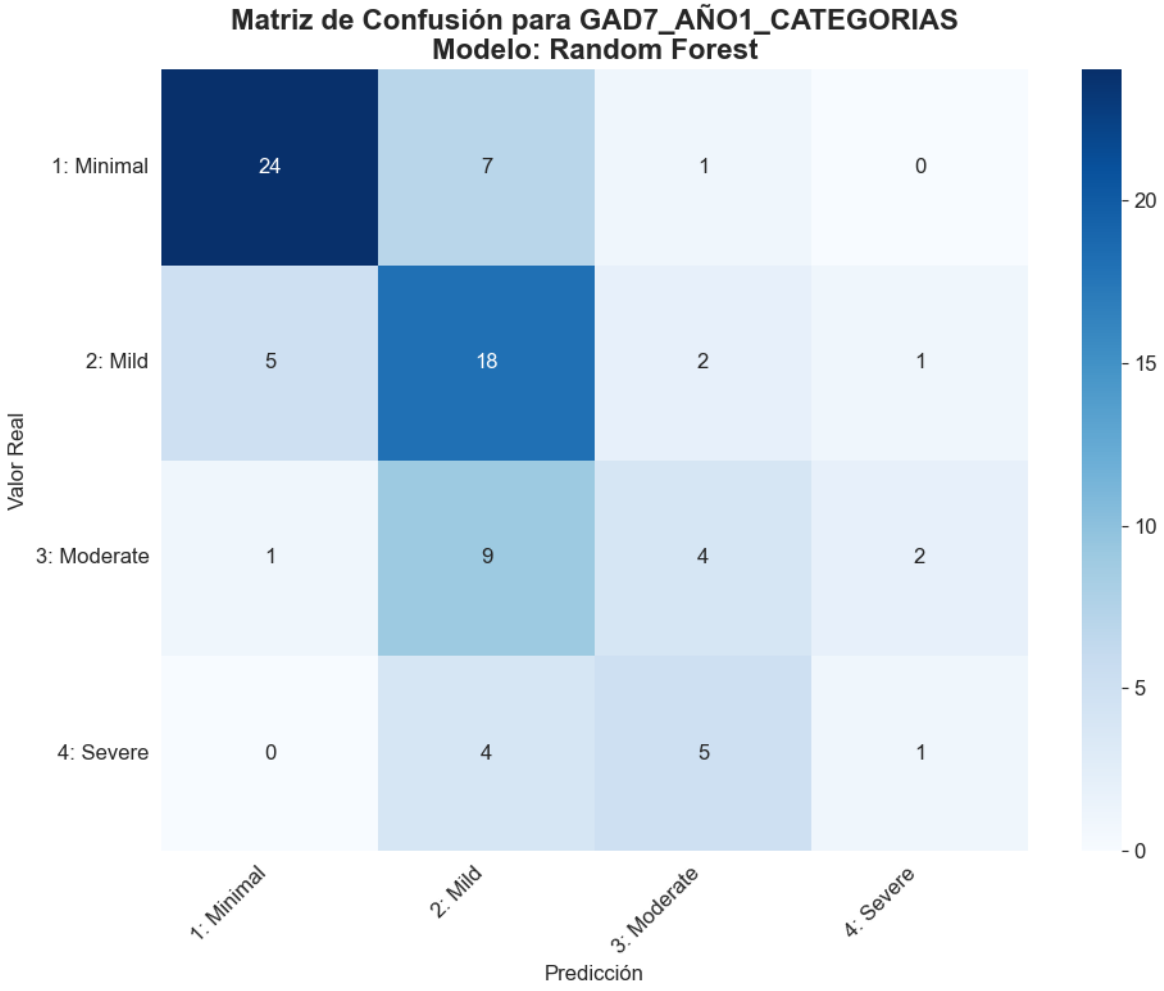


Ilustración 47 Matriz de Confusión GAD7 RF Classifier

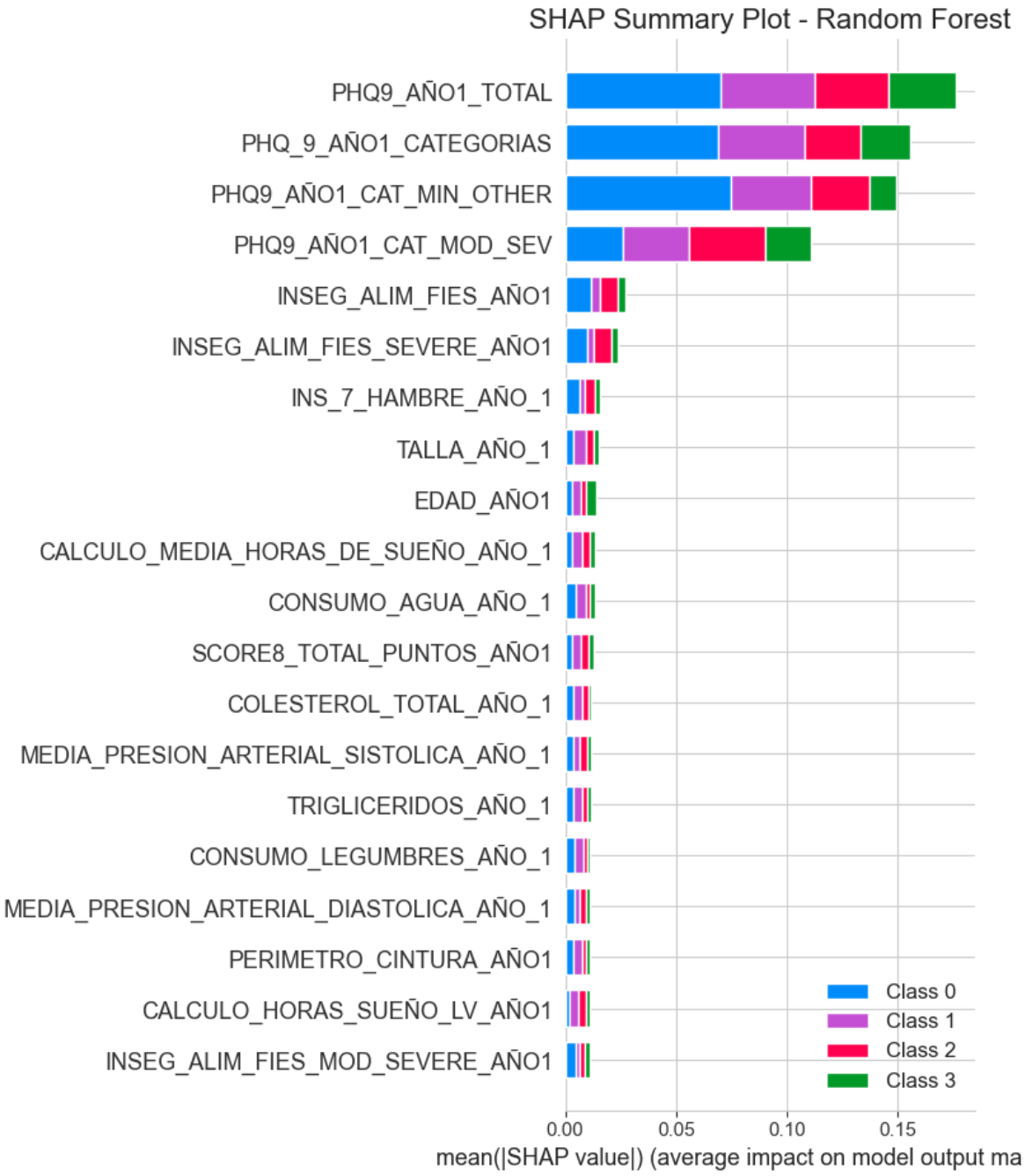


Ilustración 48 Importancia de variables GAD7 RF Classifier

5.4.3 LIGHTGBM CLASSIFIER

El siguiente paso consiste en la aplicación de algoritmos de gradient boosting, y se comienza con LightGBM Classifier por su reconocida eficiencia. La filosofía de boosting se mantiene: los árboles se construyen de forma secuencial, donde cada nuevo árbol se entrena para corregir los errores del ensamblaje anterior.

La principal adaptación para la tarea de clasificación se encuentra en la función de pérdida que se optimiza. En lugar de minimizar el error cuadrático medio, los clasificadores de boosting minimizan una función de pérdida como la pérdida logarítmica (log-loss) o la entropía cruzada binaria/multiclase. Esta función mide la divergencia entre la distribución de probabilidad predicha por el modelo y la distribución real (donde la clase correcta tiene una probabilidad de 1 y las demás de 0). Las optimizaciones propias de LightGBM, como el algoritmo basado en histogramas y el crecimiento leaf-wise, se aplican igualmente para acelerar este proceso de optimización de la pérdida logarítmica.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.5714, F1-Score (Weighted): 0.5615

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.3214, F1-Score (Weighted): 0.3021

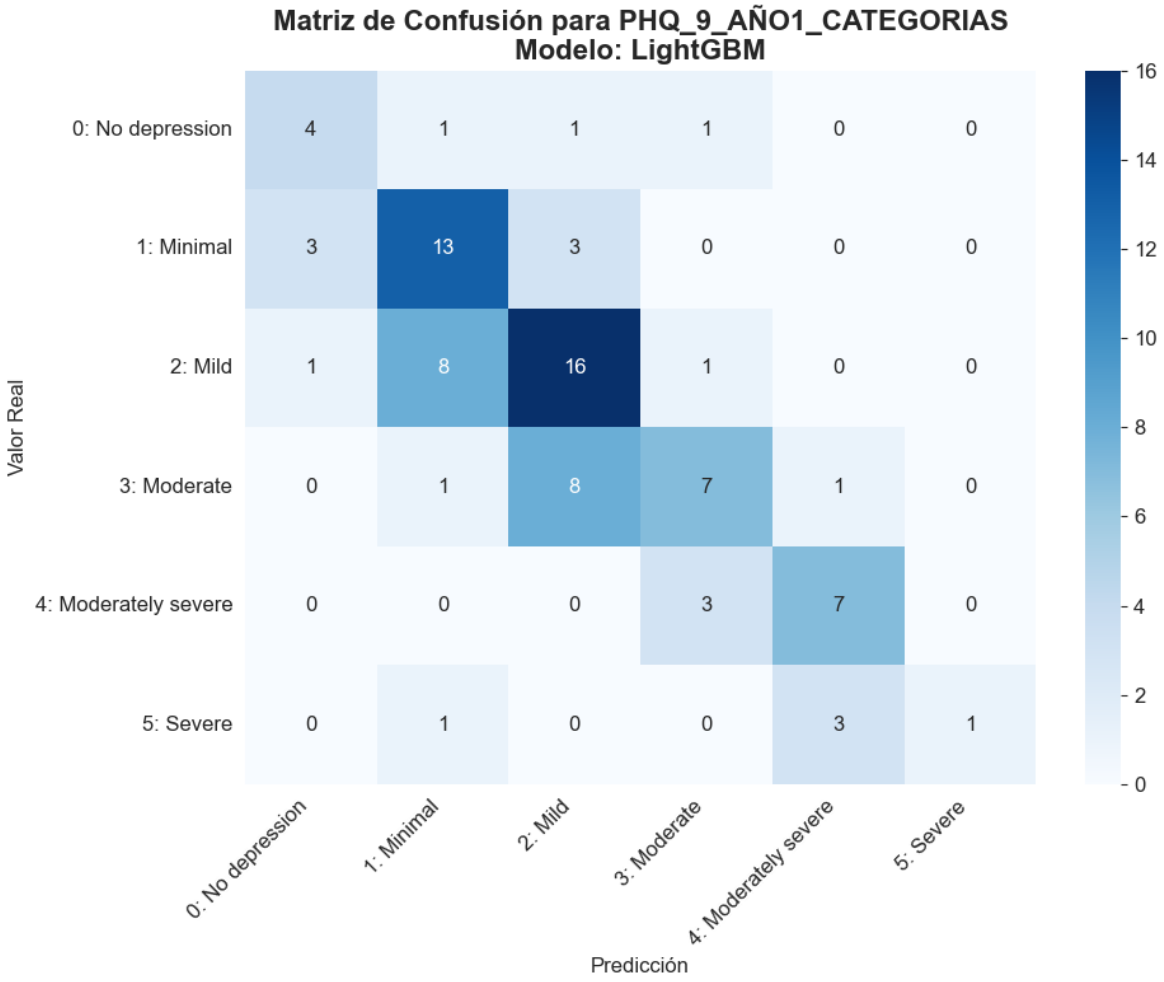


Ilustración 49 Matriz de Confusión PHQ9 LightGBM Classifier

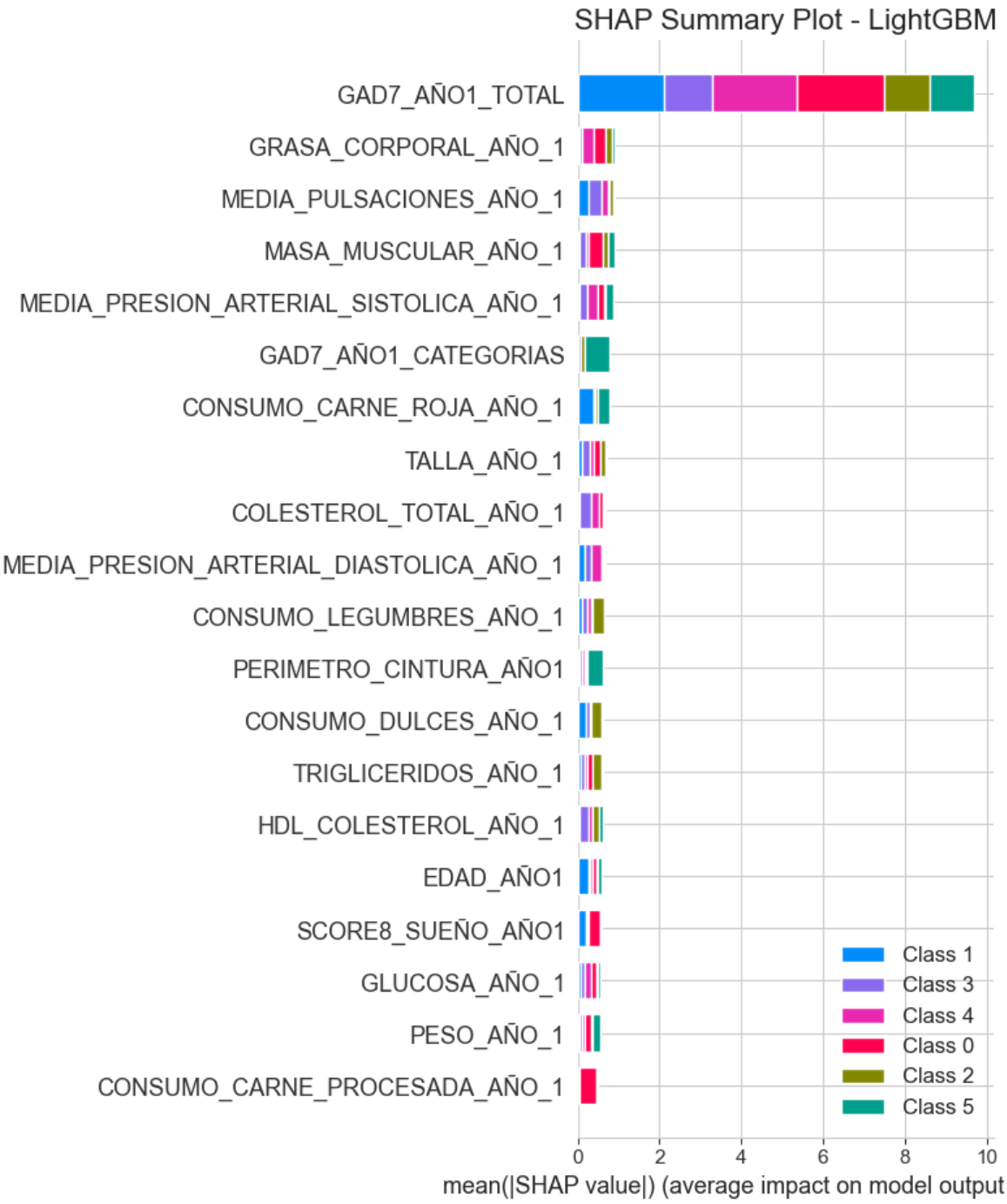


Ilustración 50 Importancia de variables PHQ9 LightGBM Classifier

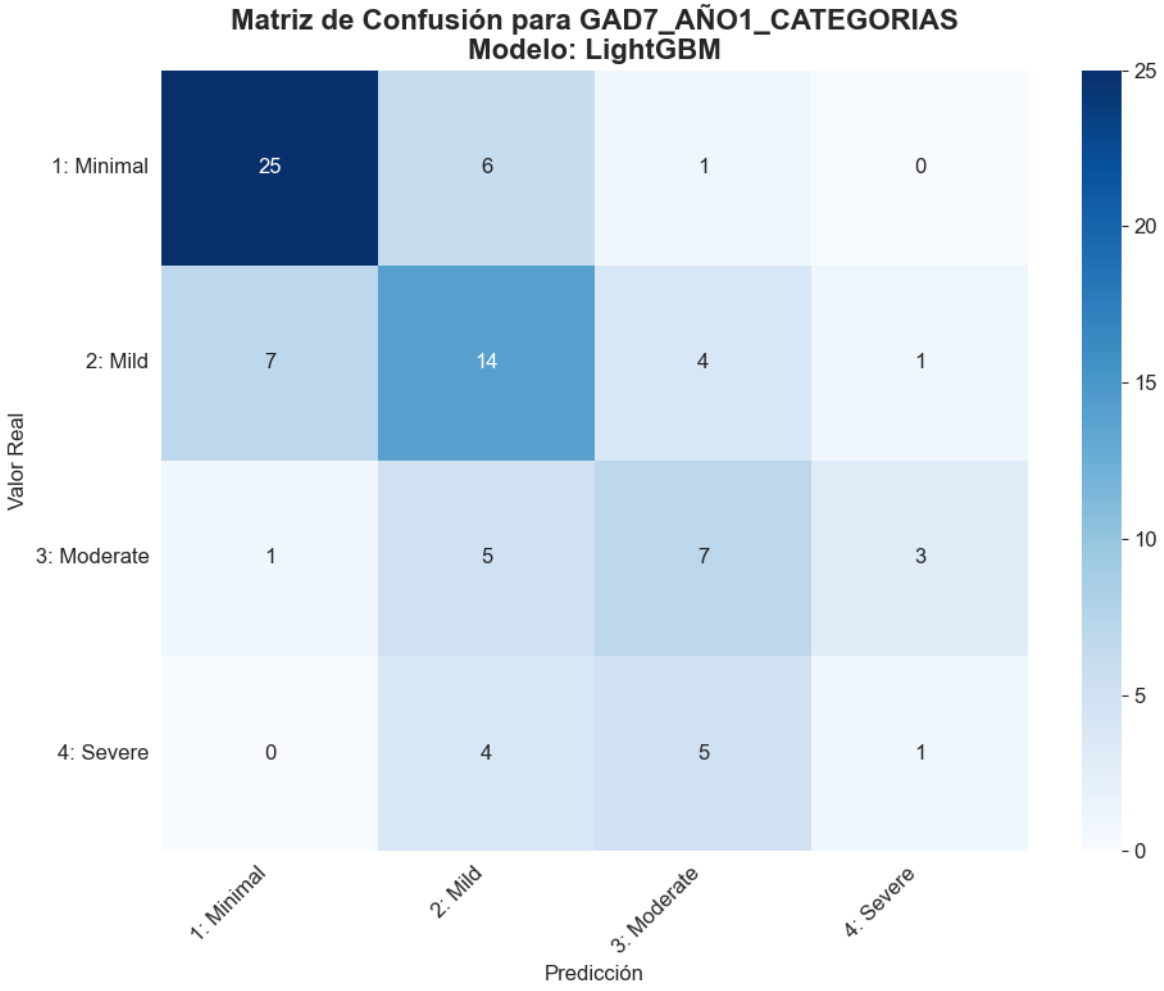


Ilustración 51 Matriz de Confusión GAD7 LightGBM Classifier

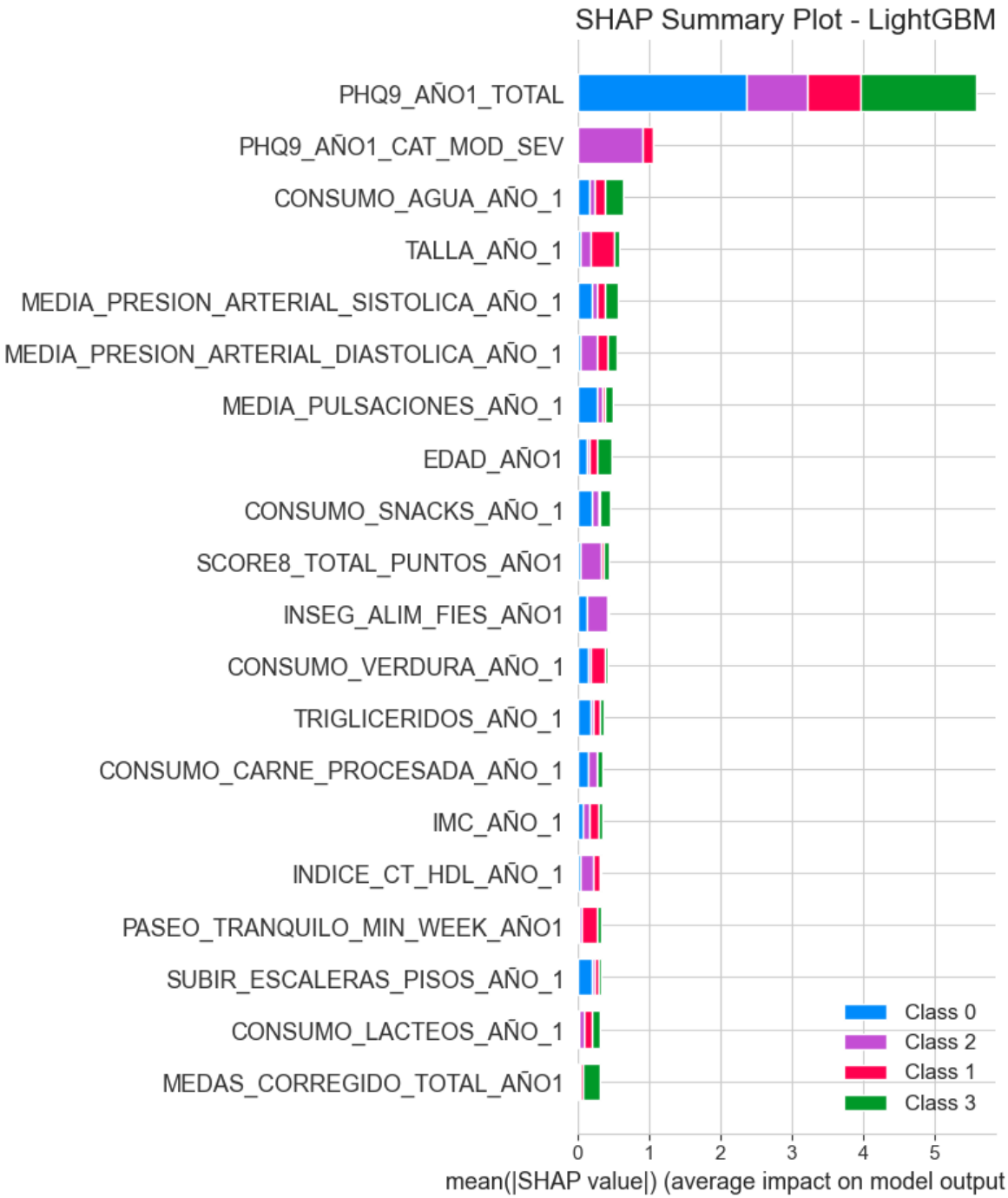


Ilustración 52 Importancia de variables GAD7 LightGBM Classifier

5.4.4 XGBOOST CLASSIFIER

Para asegurar una evaluación exhaustiva, se implementa XGBoost Classifier como benchmark de alto rendimiento frente a LightGBM. La reputación de XGBoost se extiende igualmente a los problemas de clasificación, donde su precisión y robustez son altamente valoradas.

Al igual que LightGBM, XGBoost optimiza una función de pérdida logarítmica para guiar su entrenamiento secuencial. Su ventaja competitiva a menudo radica en su formulación matemática más rigurosa, que incluye una aproximación de segundo orden de Taylor (utilizando gradiente y hessiano) para la función de pérdida y una regularización L1 y L2 integrada. Esta regularización, aplicada sobre las puntuaciones de las hojas, proporciona un control muy efectivo sobre la complejidad del modelo, resultando en una excelente capacidad de generalización y resistencia al sobreajuste, incluso en problemas de clasificación complejos.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.5714, F1-Score (Weighted): 0.5719

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2381, F1-Score (Weighted): 0.2358

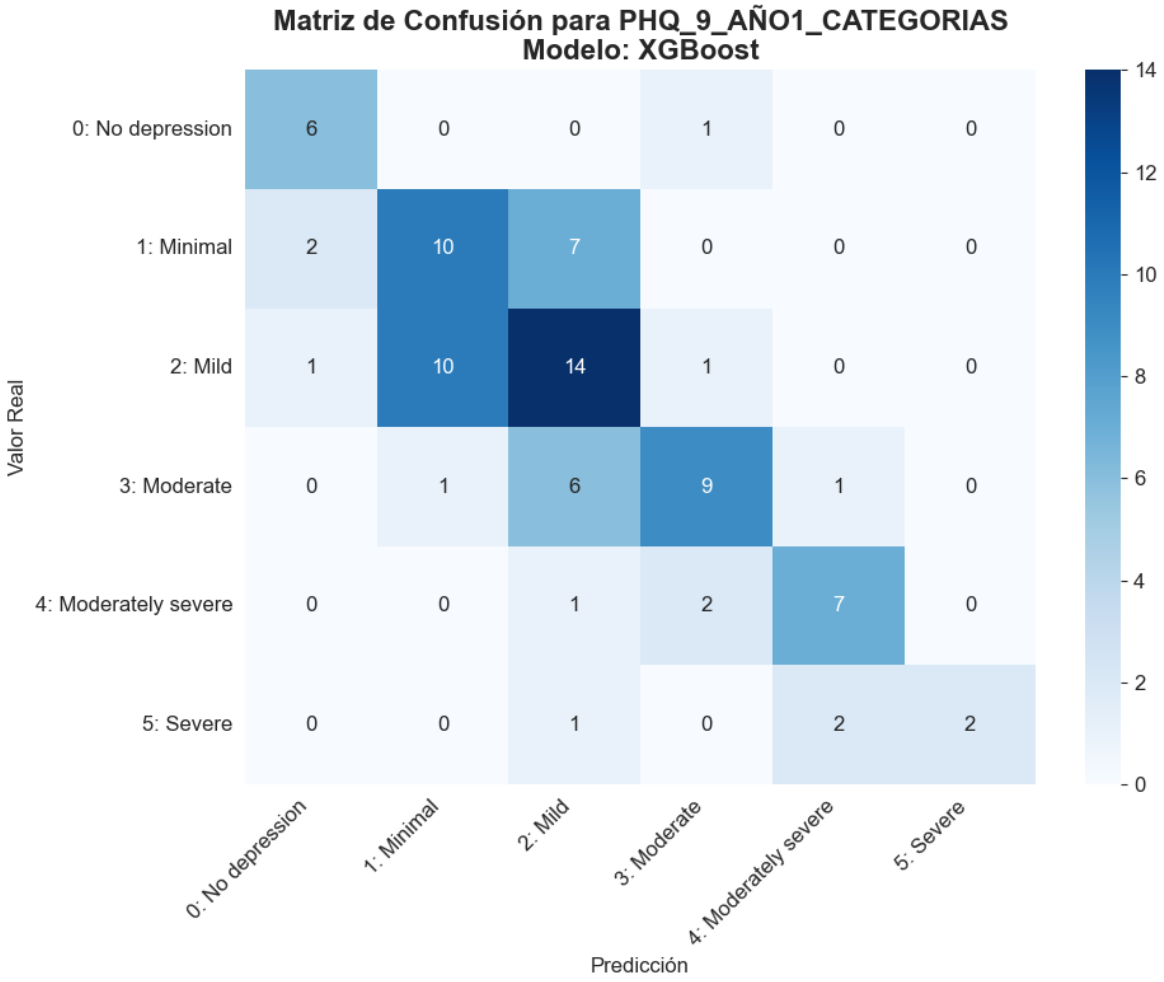


Ilustración 53 Matriz de Confusión PHQ9 XGBoost Classifier

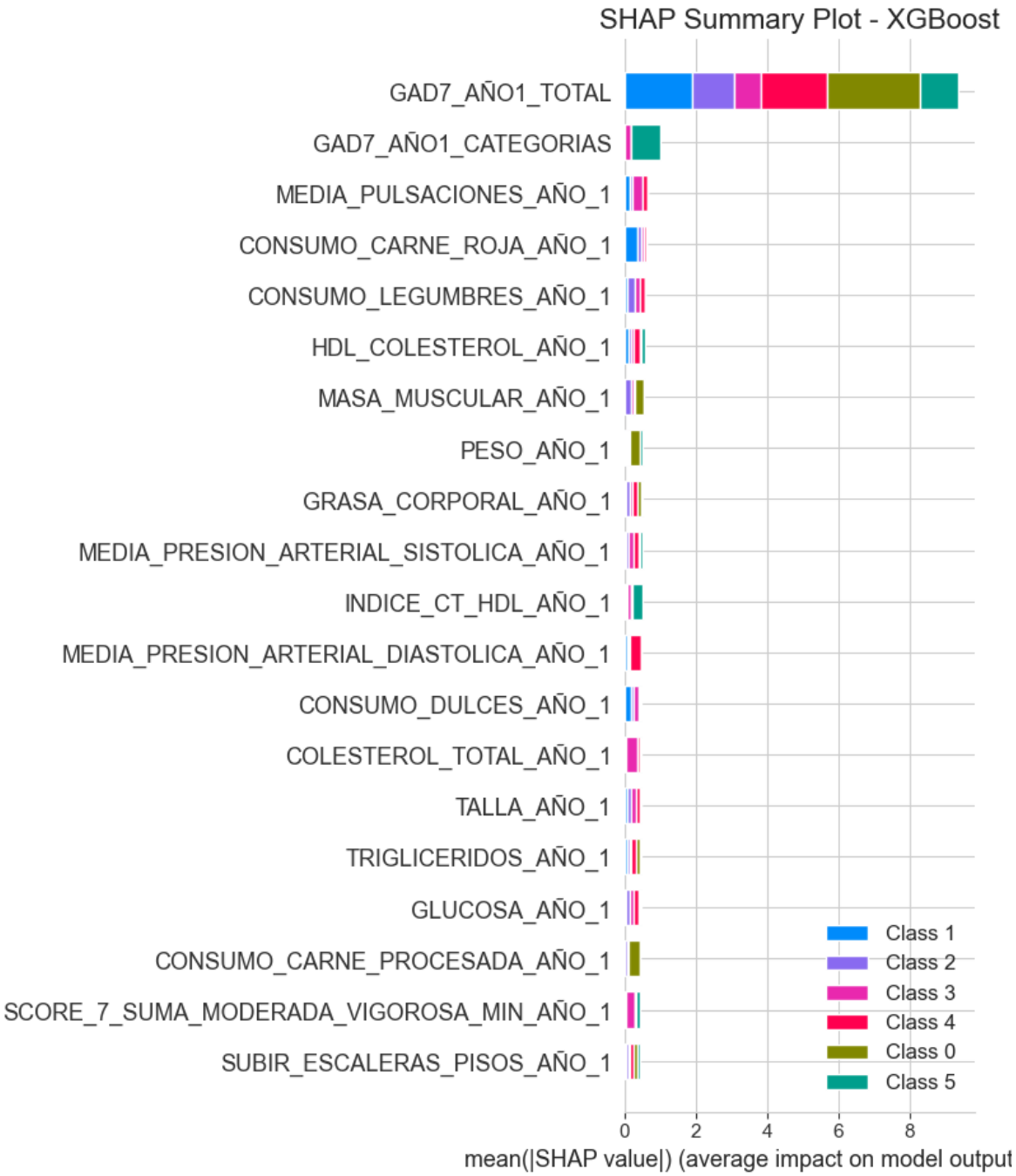


Ilustración 54 Importancia de variables PHQ9 XGBoost Classifier

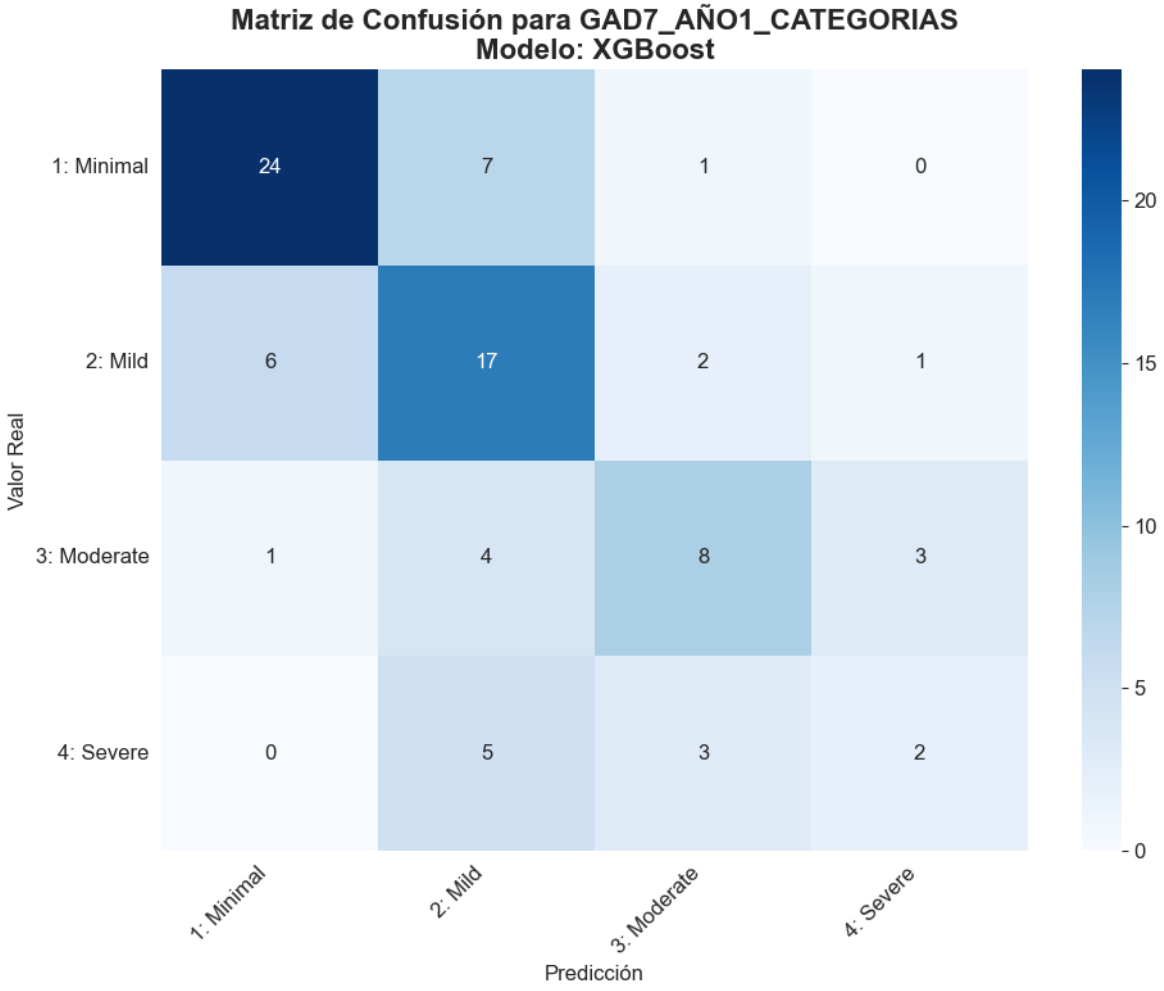


Ilustración 55 Matriz de Confusión GAD7 XGBoost Classifier

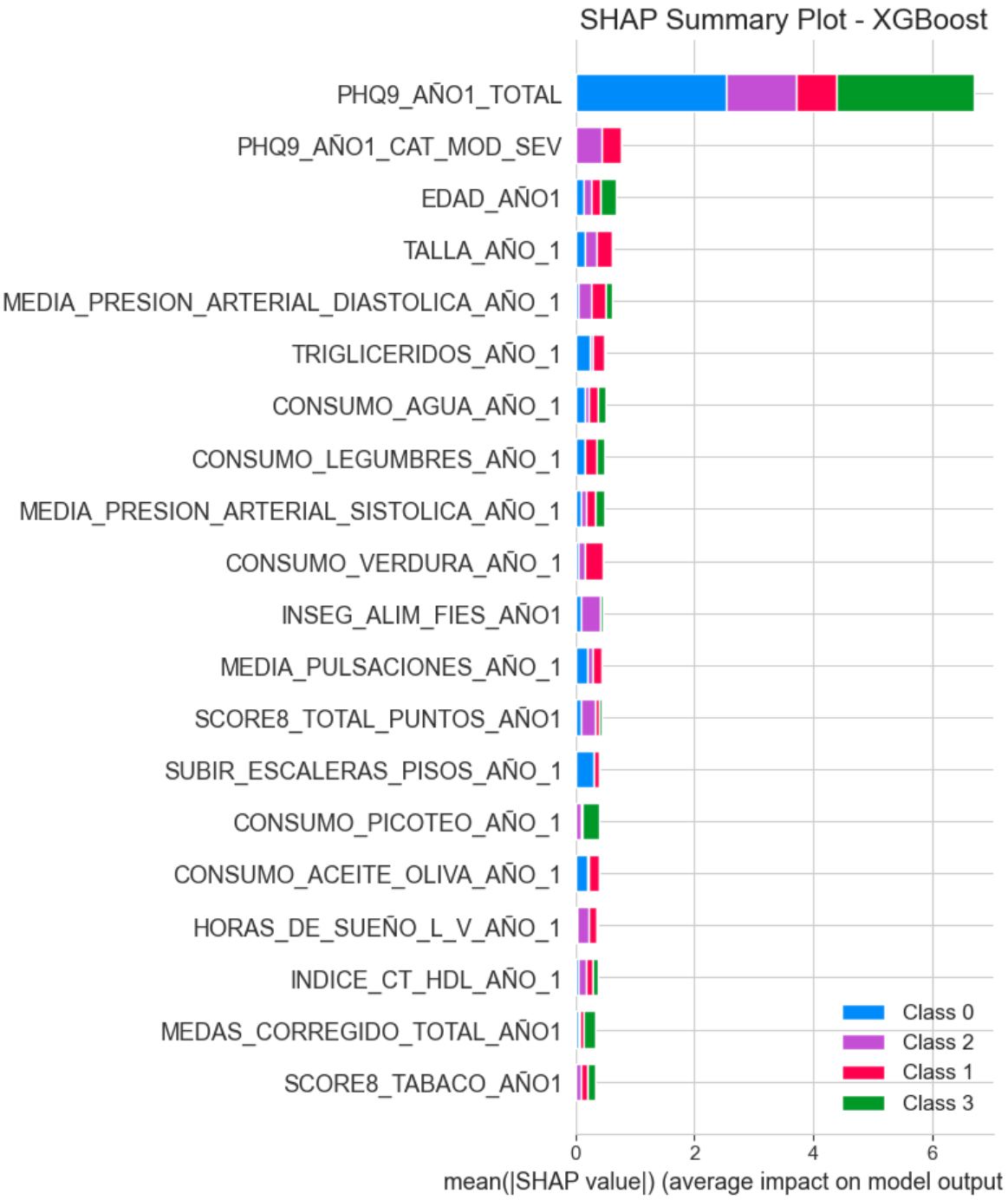


Ilustración 56 Importancia de variables GAD7 XGBoost Classifier

5.4.5 CATBOOST CLASSIFIER

El análisis de los modelos de boosting concluye con CatBoost Classifier, cuya principal ventaja es su sofisticado manejo nativo de variables categóricas. En problemas de clasificación, donde las características categóricas (como códigos postales, tipos de producto, etc.) suelen ser muy informativas, un tratamiento adecuado es fundamental.

CatBoost utiliza su técnica de Ordered Target Statistics para codificar estas variables. Este método calcula estadísticas de la variable objetivo (en este caso, relacionadas con la probabilidad de la clase) de una manera que previene la fuga de información, un riesgo común en otras técnicas de codificación. La utilización de árboles simétricos (oblivious trees) también se mantiene, actuando como un regularizador implícito que es particularmente efectivo en tareas de clasificación. Su inclusión es vital para determinar si un enfoque especializado en las características del dataset puede superar a los modelos de boosting de propósito general.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.5238, F1-Score (Weighted): 0.505

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2976, F1-Score (Weighted): 0.2777

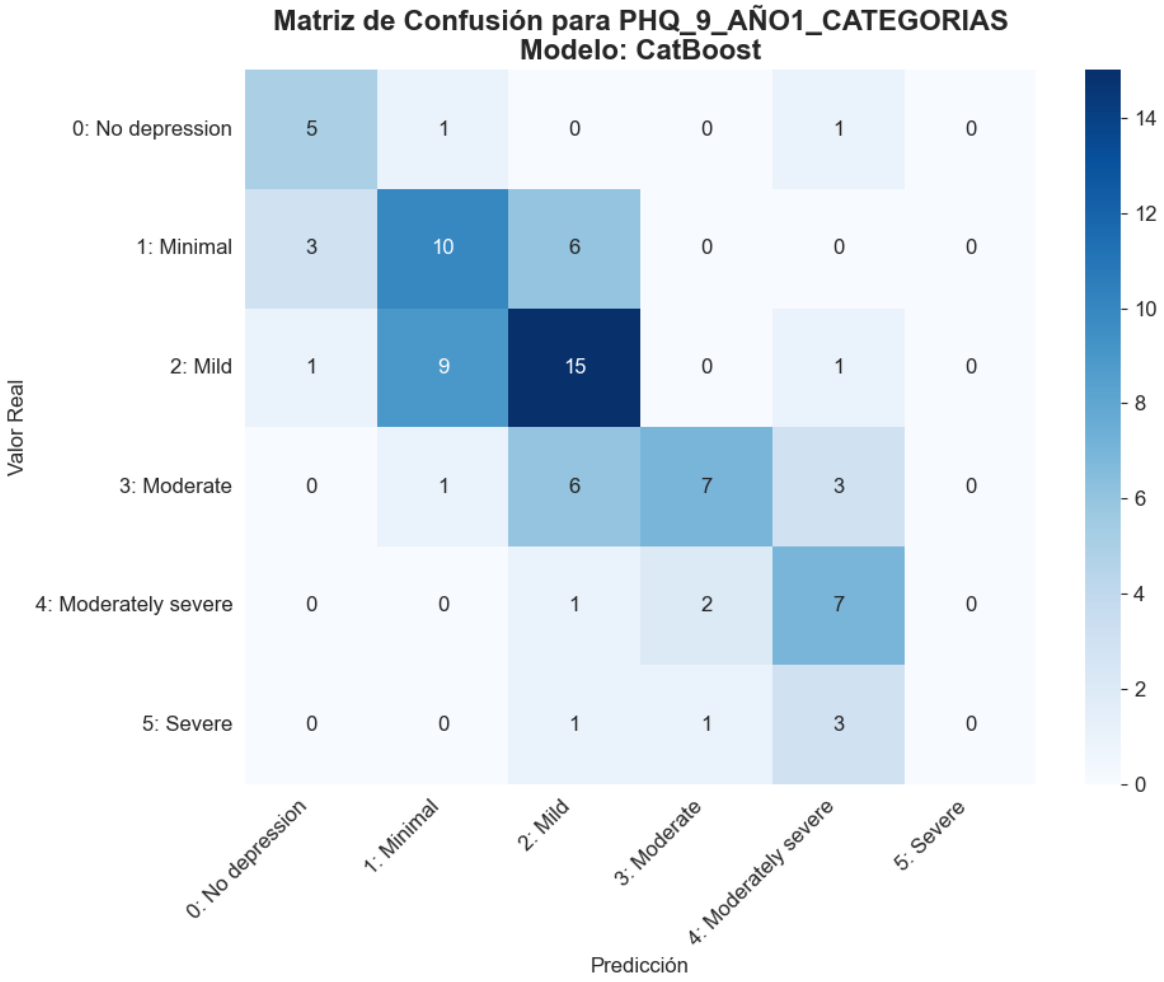


Ilustración 57 Matriz de Confusión PHQ9 CatBoost Classifier

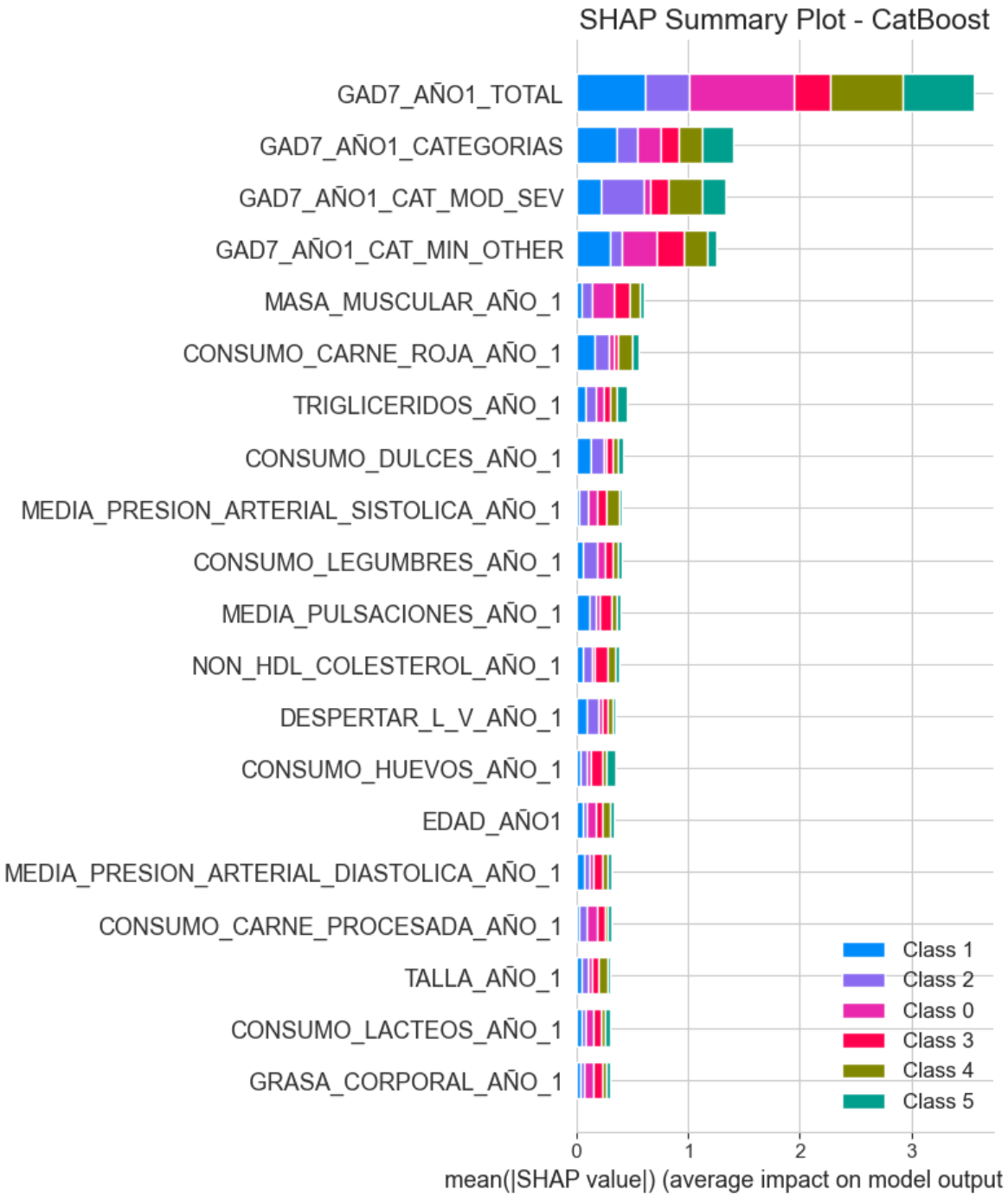


Ilustración 58 Importancia de variables PHQ9 CatBoost Classifier

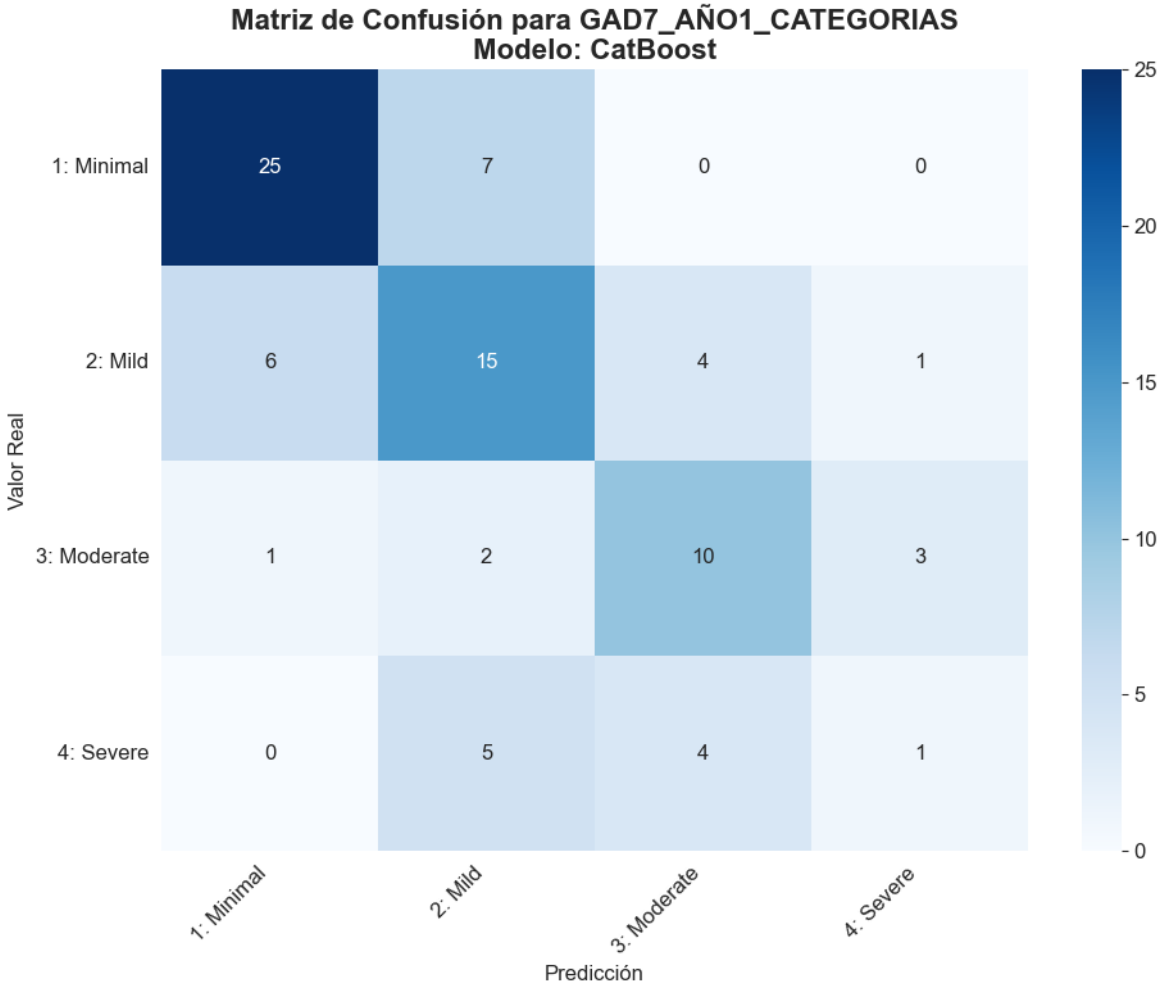


Ilustración 59 Matriz de Confusión GAD7 CatBoost Classifier

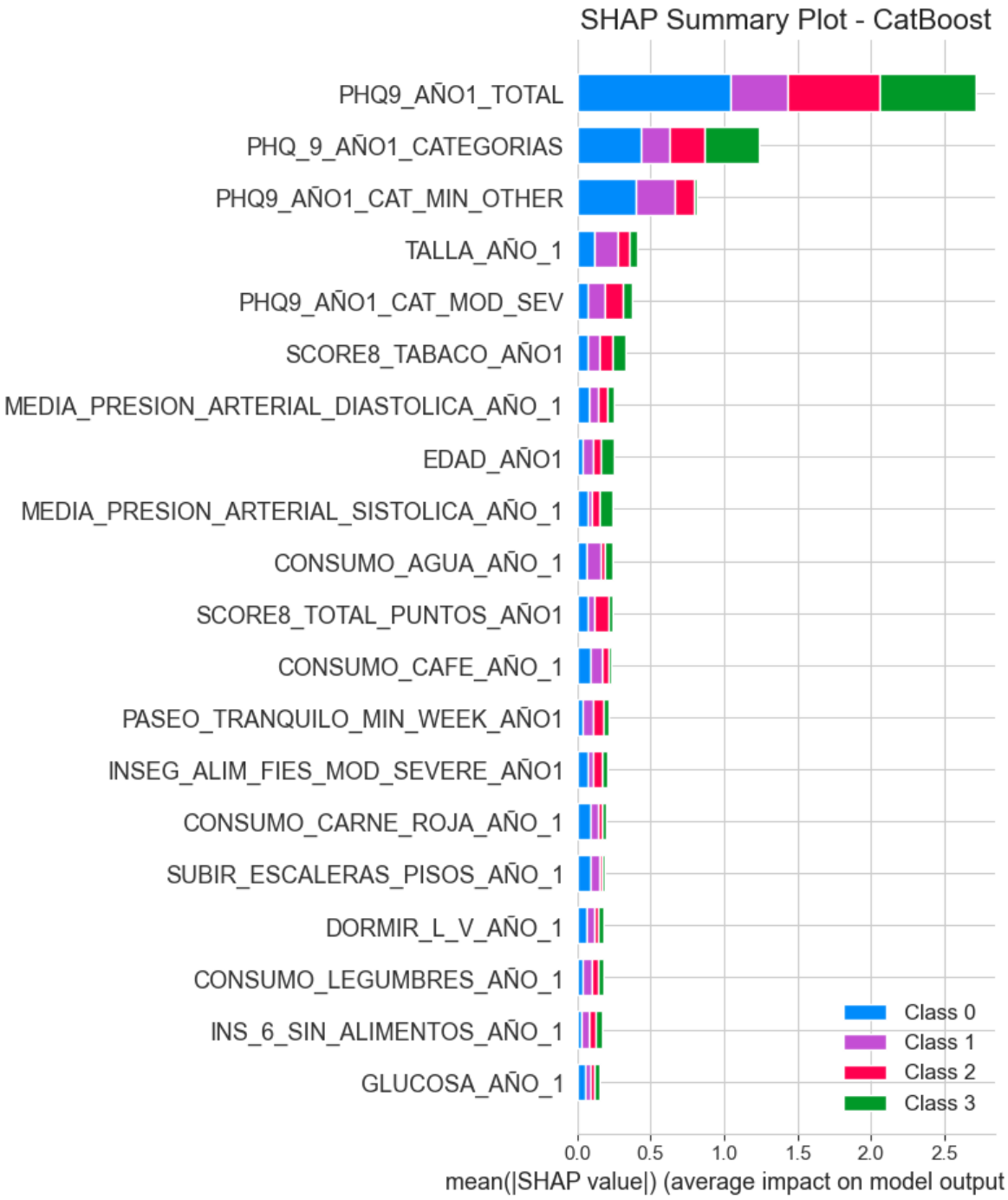


Ilustración 60 Importancia de variables GAD7 CatBoost Classifier

5.4.6 STACKING CLASSIFIER

La fase culminante para maximizar el rendimiento predictivo es la implementación del Stacking Classifier. Esta arquitectura de meta-ensamblaje busca combinar las fortalezas de los clasificadores heterogéneos entrenados previamente.

El funcionamiento es análogo al de su versión de regresión. En el Nivel 0, los modelos base (Random Forest, LightGBM, XGBoost y CatBoost) generan predicciones. Para la clasificación, estas predicciones suelen ser las probabilidades de clase para cada instancia, obtenidas mediante validación cruzada para evitar el sobreajuste. En el Nivel 1, estas probabilidades se convierten en el nuevo conjunto de características para entrenar un meta-modelo, que en este caso es la Regresión Logística. Este meta-modelo aprende la mejor forma de combinar las probabilidades de entrada para realizar una predicción final más informada y precisa. La elección de la Regresión Logística es ideal por su capacidad para modelar bien las relaciones entre estas probabilidades de entrada, que ya son características bien comportadas.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.4524, F1-Score (Weighted): 0.4319

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2857, F1-Score (Weighted): 0.2563

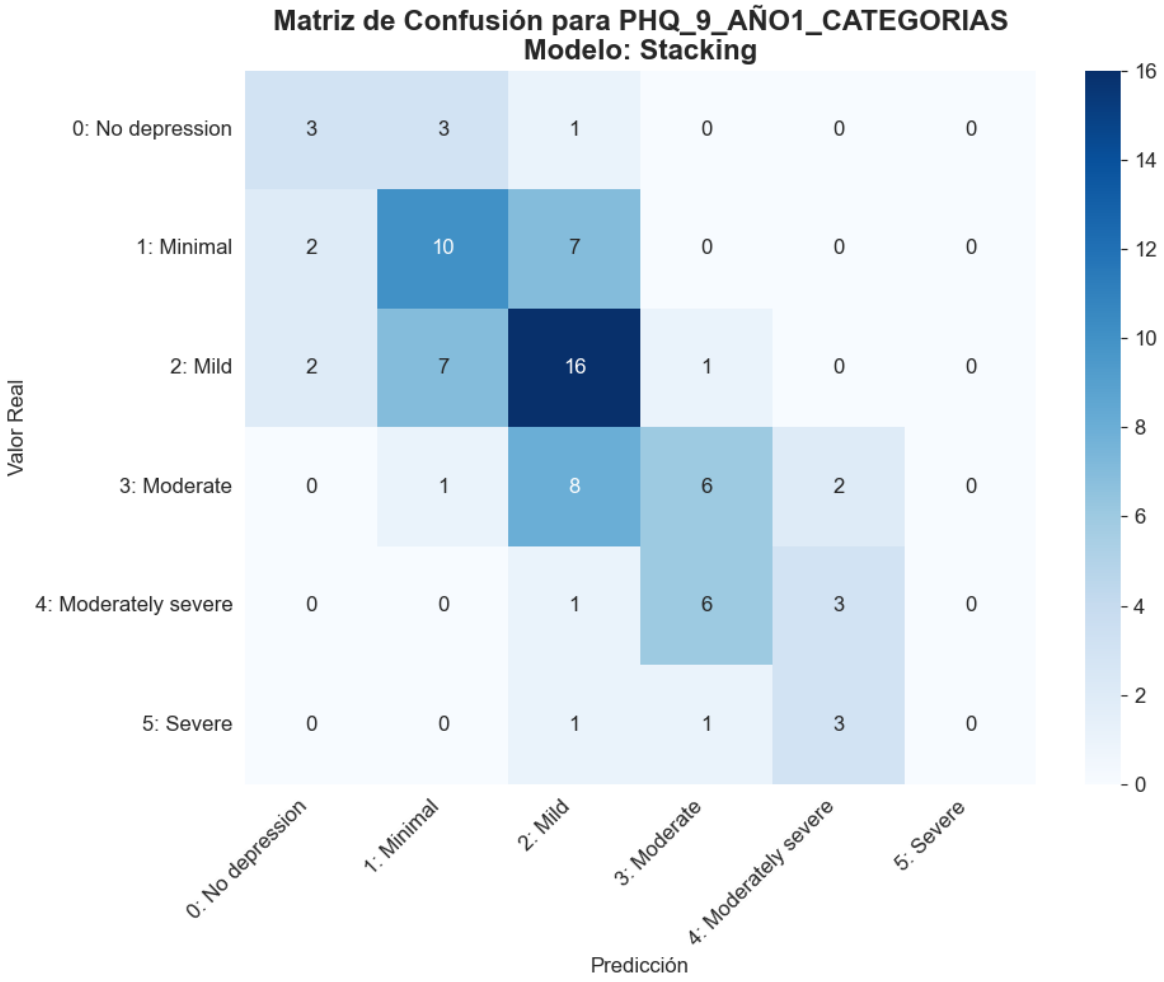


Ilustración 61 Matriz de Confusión PHQ9 Stacking Classifier

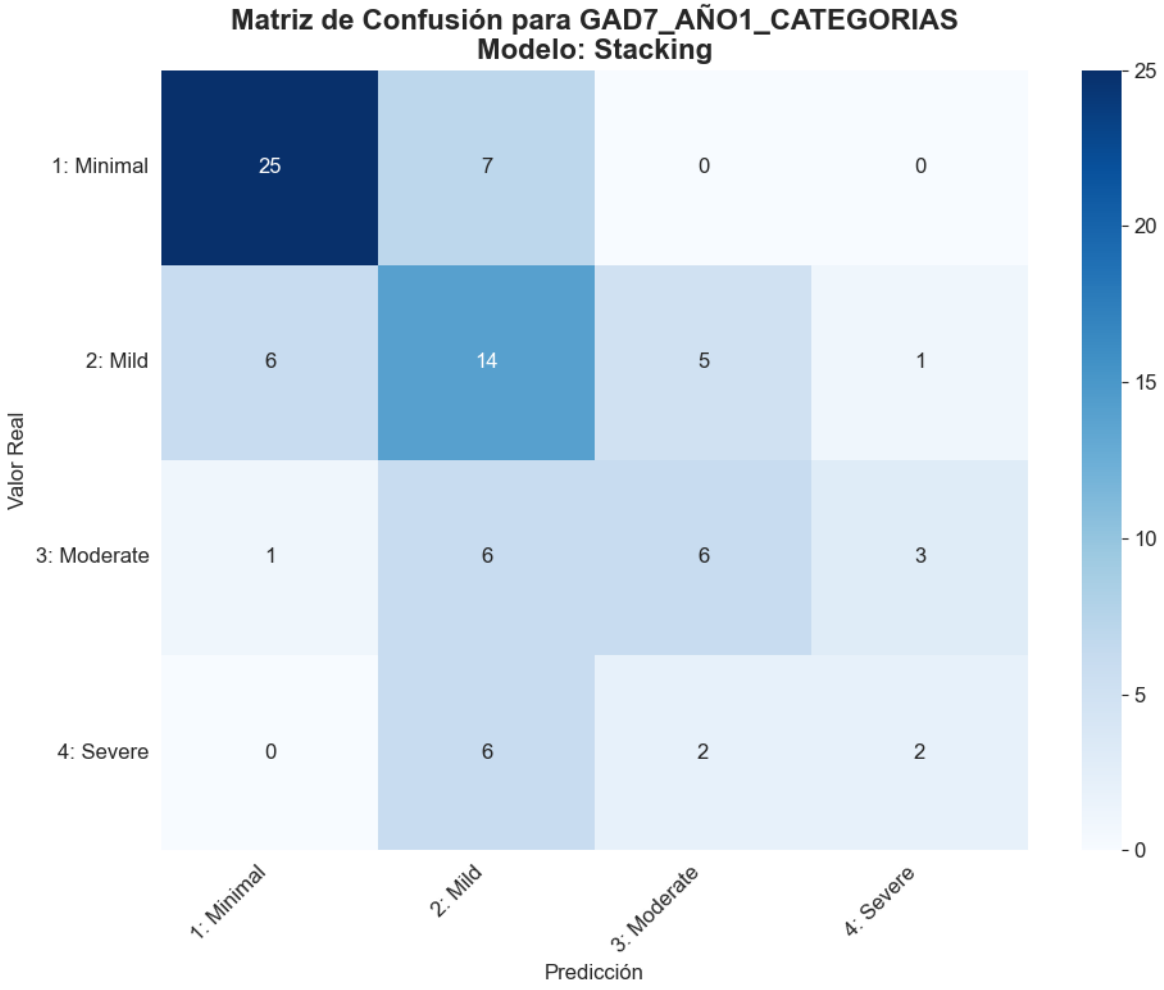


Ilustración 62 Matriz de Confusión GAD7 Stacking Classifier

5.4.7 FEDERATED LOGISTIC REGRESSION

Finalmente, el proyecto aborda el paradigma del Aprendizaje Federado para la tarea de clasificación, utilizando la Regresión Logística Federada. Este enfoque simula un escenario realista donde los datos sensibles necesarios para la clasificación no pueden ser centralizados.

El proceso es conceptualmente idéntico al del Ridge Federado. Un modelo de Regresión Logística global es distribuido a múltiples clientes. Cada cliente lo entrena con sus datos locales, optimizando los coeficientes para minimizar la pérdida logarítmica local. Posteriormente, solo los parámetros del modelo (coeficientes e intercepto) son enviados a un servidor central, donde son agregados mediante promediado para actualizar el modelo global. Este ciclo se repite, mejorando iterativamente el modelo sin compartir nunca los datos brutos.

La Regresión Logística es el modelo perfecto para esta simulación por su simplicidad paramétrica. Este experimento es crucial para evaluar el trade-off entre la precisión del clasificador y la garantía de privacidad, una consideración indispensable para la aplicación de modelos de machine learning en el mundo real.

Métricas finales en Test para predecir PHQ_9_AÑO1_CATEGORIAS: Accuracy: 0.3214, F1-Score (Weighted): 0.2621

Métricas finales en Test para predecir GAD7_AÑO1_CATEGORIAS: Accuracy: 0.2262, F1-Score (Weighted): 0.1731

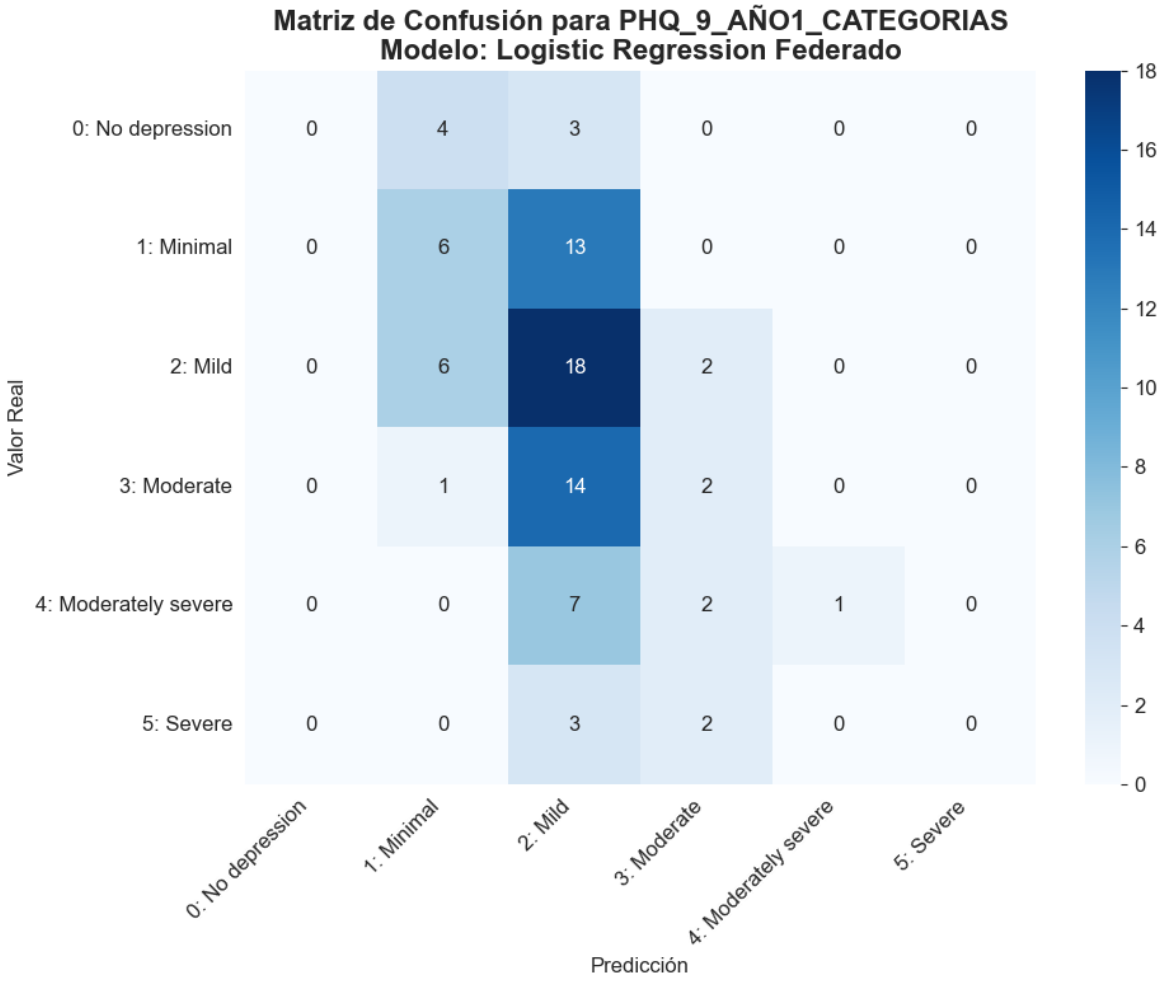


Ilustración 63 Matriz de Confusión PHQ9 Federated Logistic Regressor

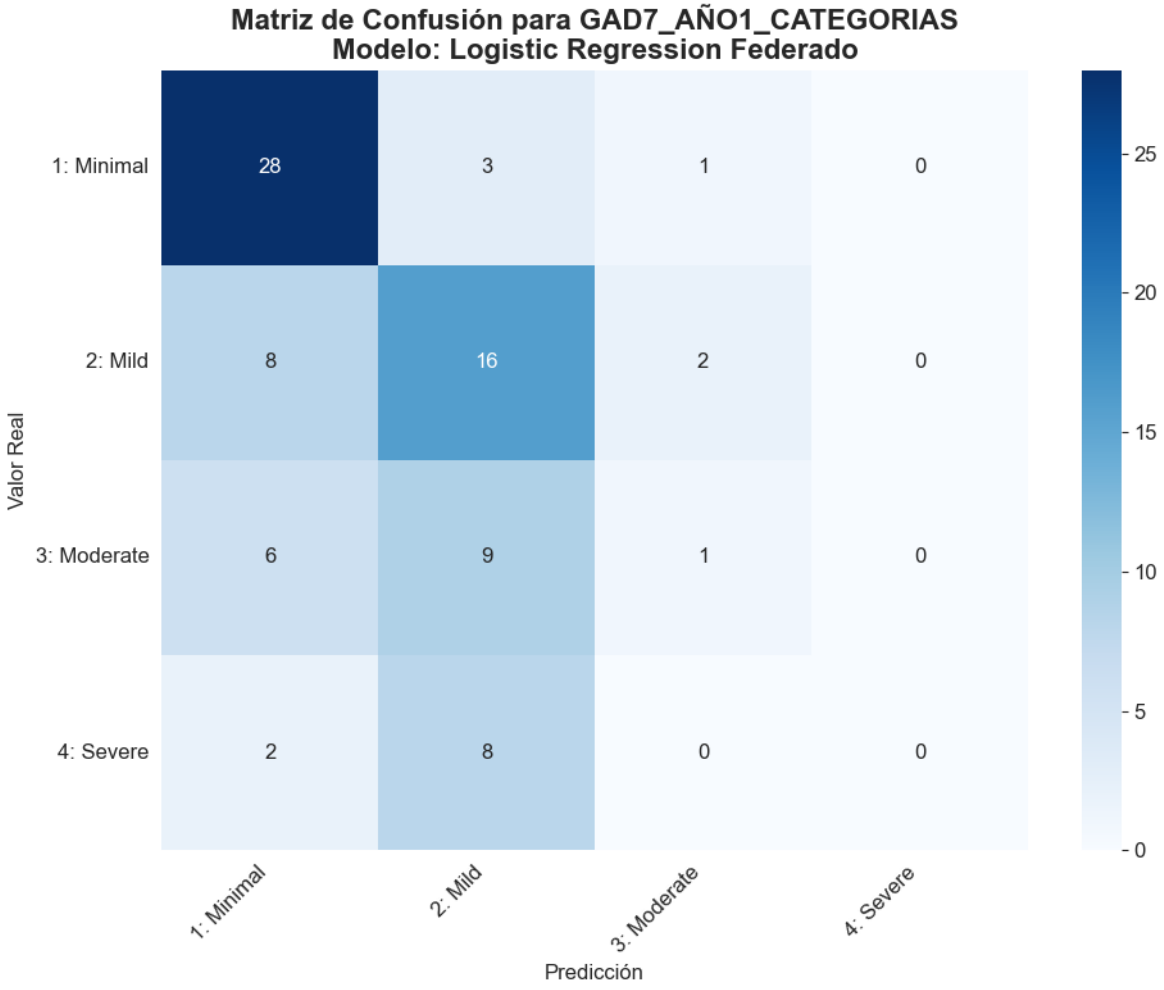


Ilustración 64 Matriz de Confusión GAD7 Federated Logistic Regressor

5.4.8 COMPARACIÓN GENERAL DE LOS MODELOS

Para facilitar un análisis final de los resultados generaremos unos gráficos y tablas que comparen todos los modelos entrenados:

Comparativa de Rendimiento Final para PHQ_9_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.6602	2.6131	3.3475
Random Forest	0.6293	2.7237	3.4964
Stacking	0.6155	2.7305	3.5609
XGBoost	0.6113	2.6624	3.5802
CatBoost	0.5970	2.8141	3.6451
Ridge Federado	0.4691	3.5604	4.1840
Ridge	0.3387	3.6274	4.6698

Tabla 6 Comparativa de Rendimiento Final PHQ9

Comparativa de Rendimiento Final para GAD_7_AÑO1_TOTAL			
Modelo	R^2	MAE	RMSE
LightGBM	0.5850	2.6441	3.4523
Random Forest	0.5751	2.5419	3.4931
Stacking	0.5681	2.5764	3.5220
CatBoost	0.5617	2.6390	3.5478
XGBoost	0.4615	2.8784	3.9327
Ridge	0.4149	3.1910	4.0992
Ridge Federado	0.4129	3.1196	4.1061

Tabla 7 Comparativa de Rendimiento Final GAD7

5.4.9 RESULTADOS ELIMINANDO LAS VARIABLES CON MAYOR IMPORTANCIA

Vista la gran correlación entre las variables de depresión y estrés, entrenaremos de nuevo los modelos prescindiendo del estudio GAD7 cuando se intente predecir el PHQ9 y viceversa, obteniendo los siguientes resultados:

Comparativa de Rendimiento Final para PHQ_9_AÑO1_TOTAL			
	R^2	MAE	RMSE
Modelo			
Ridge Federado	0.0187	4.6471	5.6883
Random Forest	0.0030	4.5772	5.7337
Stacking	-0.0154	4.6217	5.7864
LightGBM	-0.0733	4.7862	5.9490
CatBoost	-0.0774	4.7915	5.9604
XGBoost	-0.1970	4.9177	6.2825
Ridge	-0.2931	5.2822	6.5299

Tabla 8 Comparativa de Rendimiento Final PHQ9 sin GAD7

Comparativa de Rendimiento Final para GAD_7_AÑO1_TOTAL			
	R^2	MAE	RMSE
Modelo			
CatBoost	0.0592	4.2075	5.1977
LightGBM	0.0336	4.2267	5.2681
Stacking	0.0122	4.3306	5.3261
Random Forest	-0.0199	4.3728	5.4119
Ridge Federado	-0.0644	4.2595	5.5288
XGBoost	-0.1803	4.6300	5.8221
Ridge	-0.4211	4.8502	6.3883

Tabla 9 Comparativa de Rendimiento Final GAD7 sin PHQ9

Capítulo 6. ANÁLISIS DE RESULTADOS

Tras el desarrollo de los modelos predictivos y el análisis de los datos disponibles, se han obtenido varios resultados relevantes que permiten extraer conclusiones tanto a nivel técnico como clínico. A continuación, se detallan los principales hallazgos del estudio:

6.1 PRECISIÓN POR ENCIMA DEL AZAR EN UN CONTEXTO DE POCOS DATOS

Uno de los primeros aspectos a destacar es que, a pesar de trabajar con una base de datos limitada, los modelos desarrollados han sido capaces de superar el rendimiento esperado por azar. Con una precisión media cercana al 60%, los resultados indican que los modelos han logrado captar patrones reales en los datos, aunque aún lejos de alcanzar una precisión ideal para su uso clínico.

Esto sugiere que, incluso con pocas observaciones, existen señales consistentes en las variables relacionadas con la ansiedad y la depresión que pueden ser aprovechadas por los algoritmos de aprendizaje automático.

6.2 RELACIÓN ESTRECHA ENTRE ANSIEDAD Y DEPRESIÓN

Uno de los resultados más significativos ha sido la evidencia empírica de la relación entre los síntomas de ansiedad (o estrés) y los de depresión. Al eliminar las variables relacionadas con la depresión durante la predicción de la ansiedad, el rendimiento de los modelos se redujo de forma notable, y lo mismo ocurrió al predecir depresión sin contar con variables relacionadas con el estrés.

Este hallazgo no solo confirma lo que ya se conoce desde la literatura clínica —la frecuente comorbilidad entre ansiedad y depresión—, sino que además demuestra que estas relaciones pueden observarse claramente en los datos y ser detectadas por modelos computacionales.

6.3 AUSENCIA DE PREDICTORES SECUNDARIOS RELEVANTES

Uno de los objetivos del análisis era determinar si existían otros patrones o combinaciones de variables con capacidad predictiva. Para investigar esto, se aisló el efecto de las variables de ansiedad al predecir la depresión, y viceversa.

El resultado de este experimento fue determinante para comprender las limitaciones del estudio. Al neutralizar la influencia de la variable predictora principal, la capacidad de todos los modelos se desplomó. Ningún algoritmo, desde la regresión lineal hasta los ensamblajes de boosting, logró identificar una estructura o correlación alternativa en los datos restantes que permitiera construir un modelo medianamente bueno.

Por lo tanto, una de las conclusiones fundamentales de este trabajo es que, si bien la interdependencia entre ansiedad y depresión es muy alta, no se ha logrado encontrar ninguna otra correlación secundaria de interés que sea lo suficientemente fuerte como para sostener un modelo predictivo útil. Esto sugiere que, con las variables disponibles no se han encontrado relaciones fuertes con las enfermedades a predecir.

6.4 MEJORA ESPERADA DEL MODELO CON MÁS DATOS

A lo largo del proyecto se ha podido observar que, a medida que se incrementa el número de observaciones (aunque sea de forma limitada), el rendimiento de los modelos tiende a mejorar. Esta tendencia sugiere que con una base de datos más amplia y representativa, los modelos podrían alcanzar niveles de precisión mucho más altos.

Esto refuerza la idea de que este tipo de modelos son escalables, y que contar con más datos —especialmente si proceden de distintas fuentes y contextos— podría permitir una predicción mucho más robusta y generalizable.

6.5 UTILIDAD DEL APRENDIZAJE FEDERADO

El uso del aprendizaje federado en este trabajo ha sido especialmente relevante, ya que ha permitido entrenar modelos sin necesidad de centralizar los datos, lo cual tiene un valor añadido en entornos clínicos donde la privacidad del paciente es una prioridad. A pesar de esta limitación, los modelos federados han conseguido rendimientos comparables a los modelos tradicionales centralizados.

Este resultado demuestra que es posible construir modelos útiles y respetuosos con la confidencialidad, y que el aprendizaje federado no solo es una solución viable para problemas técnicos, sino también una herramienta valiosa para la investigación ética en salud mental.

Además, se ha observado que el aprendizaje federado puede ser útil no solo para entrenar modelos, sino también para explorar relaciones entre variables sensibles sin acceder directamente a los datos originales, lo cual amplía aún más sus aplicaciones potenciales.

6.6 VALIDACIÓN DE HIPÓTESIS CLÍNICAS DESDE UN ENFOQUE COMPUTACIONAL

Por último, el trabajo ha servido como una pequeña demostración de cómo es posible validar, desde un punto de vista computacional, principios ya establecidos en el ámbito clínico. La relación entre ansiedad y depresión, tradicionalmente estudiada desde una perspectiva psicológica o psiquiátrica, ha podido ser observada y cuantificada mediante modelos de machine learning, lo cual abre nuevas posibilidades para el estudio y la comprensión de los trastornos mentales.

6.7 RESUMEN DEL ANÁLISIS

En conjunto, los resultados obtenidos permiten concluir que, aunque se ha trabajado con recursos limitados, el enfoque utilizado ha sido capaz de identificar relaciones relevantes y

aportar evidencia útil. El proyecto pone de manifiesto que técnicas como el aprendizaje federado tienen un gran potencial para estudiar la salud mental de forma respetuosa con la privacidad y que, con más datos y una mayor diversificación, los modelos podrían convertirse en herramientas valiosas para la prevención y detección temprana de trastornos como la ansiedad y la depresión.

Capítulo 7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1 CONCLUSIONES

Este trabajo fin de grado tenía como objetivo principal explorar hasta qué punto es posible predecir síntomas de ansiedad y depresión mediante técnicas de aprendizaje automático, utilizando una base de datos reducida y aplicando métodos que preserven la privacidad de los pacientes, como el aprendizaje federado. Tras varios meses de desarrollo, experimentación y análisis, se puede afirmar que el proyecto ha cumplido con sus objetivos fundamentales y ha ofrecido resultados valiosos tanto desde un punto de vista técnico como clínico.

A lo largo del trabajo se ha conseguido entrenar diferentes modelos predictivos que, incluso con una cantidad de datos limitada, han logrado obtener resultados significativamente superiores al azar. Esto es especialmente importante en el ámbito de la salud mental, donde muchas veces no se dispone de grandes bases de datos por motivos de privacidad o confidencialidad, y donde cada pequeño avance puede tener un impacto real en la forma en que se estudian y abordan los trastornos psicológicos.

Uno de los hallazgos más relevantes ha sido la clara interdependencia entre los síntomas de ansiedad (o estrés) y los de depresión. Al eliminar del conjunto de variables aquellas relacionadas con uno de estos dos trastornos, el rendimiento del modelo para predecir el otro disminuye de forma notable. Este resultado no solo refuerza lo que ya se conoce desde la literatura clínica (es decir, que ansiedad y depresión suelen aparecer juntas o compartir síntomas) sino que además lo confirma a través de una metodología automatizada, basada en datos. En este sentido, el proyecto ha permitido validar empíricamente, desde una aproximación computacional, un principio ampliamente aceptado en psiquiatría.

Otra de las principales aportaciones del proyecto ha sido la implementación y prueba del aprendizaje federado. A pesar de ser una tecnología emergente, se ha demostrado su utilidad

práctica incluso en un entorno simulado. Los modelos entrenados bajo esta modalidad han sido capaces de mantener un rendimiento comparable al de los modelos centralizados, con la ventaja adicional de que nunca se ha accedido directamente a los datos, lo cual resulta especialmente relevante cuando se trata de información sensible como es la relacionada con la salud mental. Esto demuestra que es posible realizar investigaciones útiles sin comprometer la privacidad de los participantes, algo que no solo es deseable, sino que será cada vez más necesario en el futuro.

En conjunto, este trabajo pone de manifiesto que, incluso con recursos limitados, es posible construir modelos con cierto valor predictivo, extraer relaciones clínicas relevantes y experimentar con técnicas punteras como el aprendizaje federado. Si bien queda mucho camino por recorrer, especialmente en lo que respecta a la precisión y aplicabilidad real de estos modelos, se ha dado un primer paso hacia un enfoque más ético, automatizado y respetuoso con la privacidad en el estudio de los trastornos mentales.

7.2 TRABAJOS FUTUROS

A partir de los resultados obtenidos, se abren múltiples vías para continuar y ampliar este proyecto, tanto a nivel técnico como clínico. A continuación se enumeran algunas de las más prometedoras:

Aumento y diversificación del conjunto de datos: Uno de los principales desafíos ha sido el reducido tamaño de la base de datos utilizada. Una línea clara de mejora pasa por trabajar con un número mayor de participantes, idealmente recogidos en distintos contextos o países, lo cual permitiría no solo mejorar la precisión de los modelos, sino también su capacidad de generalización.

Validación frente a diagnósticos clínicos: Aunque los resultados han sido prometedores, sería fundamental contrastar las predicciones de los modelos con evaluaciones realizadas por profesionales de la salud mental. Esta validación permitiría comprobar si los patrones identificados por los modelos se corresponden con casos reales y clínicamente relevantes.

Exploración de modelos más avanzados: En este trabajo se han utilizado modelos relativamente sencillos, adecuados al tamaño del dataset. Sin embargo, con más datos, sería posible probar algoritmos más complejos como redes neuronales profundas, modelos secuenciales o incluso enfoques multimodales que combinen datos psicológicos con datos fisiológicos, de texto o de comportamiento digital.

Aplicación real del aprendizaje federado: Aunque se ha simulado un entorno de aprendizaje federado en una sola máquina, un paso lógico a futuro sería implementar esta tecnología en un entorno real, distribuido entre distintas instituciones sanitarias o educativas. Esto permitiría comprobar su viabilidad en condiciones reales y avanzar hacia soluciones prácticas y escalables.

Estudios longitudinales y predicción temporal: La salud mental es dinámica, y poder anticipar cómo evolucionan los síntomas a lo largo del tiempo sería de enorme utilidad. Para ello sería necesario contar con bases de datos longitudinales, que permitan entrenar modelos capaces no solo de clasificar un estado emocional actual, sino de predecir el riesgo futuro de desarrollar un trastorno o sufrir una recaída.

Incorporación de variables externas: Otra mejora interesante sería introducir variables contextuales como eventos vitales, calidad del sueño, hábitos digitales o niveles de actividad física, que podrían enriquecer el análisis y mejorar la capacidad predictiva de los modelos.

Desarrollo de herramientas prácticas: A largo plazo, una posible aplicación sería el desarrollo de herramientas digitales que puedan utilizar estos modelos para realizar cribados o seguimientos de manera respetuosa y no invasiva, ya sea en entornos clínicos, educativos o laborales.

En definitiva, este proyecto ha servido como punto de partida para mostrar que es posible utilizar inteligencia artificial, y en particular aprendizaje federado, para estudiar fenómenos complejos como la ansiedad y la depresión. Con más datos, más validaciones y un trabajo interdisciplinar, este enfoque podría convertirse en una herramienta complementaria y valiosa en la detección, prevención y comprensión de los trastornos de salud mental.

Capítulo 8. BIBLIOGRAFÍA

[4] IMIBIC. (2024). *IMIBIC participa en un proyecto internacional que aplicará Inteligencia Artificial para mejorar el diagnóstico y tratamiento del cáncer*. https://www.imibic.org/noticia/340

[5] Google Research. (s. f.). *Federated learning for mobile keyboard prediction*. 2018. https://research.google/pubs/federated-learning-for-mobile-keyboard-prediction-2/

[6] Heaven, W. D. (2019). Apple is using AI to personalize Siri without threatening your privacy. MIT Technology Review. https://www.technologyreview.com/2019/12/11/131629/apple-ai-personalizes-siri-federated-learning/

[7] De Fauw, J., Ledsam, J. R., Romera-Paredes, B., Nikolov, S., Tomasev, N., Blackwell, S., ... & Hassabis, D. (2018). Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature Medicine*, 24(9), 1342-1350. https://www.nature.com/articles/s41591-018-0107-6

[8] Microsoft Research. (s.f.). *Medical Image Analysis*. Microsoft. https://www.microsoft.com/en-us/research/project/medical-image-analysis/

[9] NVIDIA. (s. f.). *Transforming AI Healthcare with Federated Learning*. https://developer.nvidia.com/blog/transforming-ai-healthcare-with-federated-learning/

[10] Owkin. (s. f.). *Owkin's federated research network powers cutting-edge AI*. https://www.owkin.com/federated-research-network

[11] Moore, J., Collins, T., Chen, H., et al. (2023). Federated learning in drug discovery and healthcare: Applications, challenges, and future directions. Journal of Chemical Information and Modeling. https://doi.org/10.1021/acs.jcim.3c00799

[12] Capgemini. (2022). *Tres hospitales españoles utilizan Inteligencia Artificial para acelerar la detección de COVID-19 preservando la privacidad*. https://www.capgemini.com/es-es/wp-content/uploads/sites/16/2022/12/NP_-Tres-hospitales-espanoles-utilizan-Inteligencia-Artificial-para-acelerar-la-deteccion-de-COVID-19-preservando-la-privacidad.pdf

[13] Khalil, SS, Tawfik, NS y Spruit, M. Exploración del potencial del aprendizaje federado en la investigación sobre salud mental: una revisión sistemática de la literatura. Appl Intell 54 , 1619–1636 (2024). https://doi.org/10.1007/s10489-023-05095-1

[14] Li, J., Jiang, M., Qin, Y. et al. Intelligent depression detection with asynchronous federated optimization. Complex Intell. Syst. 9, 115–131 (2023). https://doi.org/10.1007/s40747-022-00729-2

[15] Rieke, N., Hancox, J., Li, W. et al. The future of digital health with federated learning. npj Digit. Med. 3, 119 (2020). https://doi.org/10.1038/s41746-020-00323-1

[16] Kroenke K, Wu J, Yu Z, Bair MJ, Kean J, Stump T, Monahan PO. Patient Health Questionnaire Anxiety and Depression Scale: Initial Validation in Three Clinical Trials. Psychosom Med. 2016 Jul-Aug;78(6):716-27. doi: 10.1097/PSY.0000000000000322. PMID: 27187854; PMCID: PMC4927366. https://pubmed.ncbi.nlm.nih.gov/27187854/

[17] Esther Porras-Pérez, Patricia J. Peña-Orihuela, Alberto Díaz-Cáceres, Alejandro Serrán-Jiménez, Javier Arenas-Montes, Antonio P. Arenas-de Larriva, Juan F. Alcala-Diaz, Fan-Yun Lan, Javier Delgado-Lista, Elena M. Yubero-Serrano, Jose M. Ordovas, Juan L. Romero-Cabrera, Pablo Pérez-Martínez, Health Inequity in Metabolic Disorders and Lifestyle Behaviors in Under-resourced Adolescents: A cross-sectional study from E-DUCASS Program, American Journal of Preventive Cardiology, (101012), (2025). https://doi.org/10.1016/j.ajpc.2025.101012

ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS

El presente trabajo, desarrollado en colaboración con el Instituto Maimónides de Investigación Biomédica de Córdoba (IMIBIC), se centra en la predicción de niveles de estrés y ansiedad en una población minoritaria y desfavorecida de Córdoba, mediante técnicas de aprendizaje federado. Esta metodología permite preservar la privacidad de los datos sensibles al evitar la centralización de la información, respetando así principios éticos fundamentales en el manejo de datos clínicos.

Este enfoque se encuentra alineado con varios de los Objetivos de Desarrollo Sostenible (ODS) propuestos por la Agenda 2030 de las Naciones Unidas. En primer lugar, contribuye al ODS 3: Salud y bienestar, al promover la detección temprana y el seguimiento de trastornos relacionados con la salud mental, en especial en colectivos vulnerables que habitualmente enfrentan barreras para acceder a servicios sanitarios adecuados. Al generar herramientas predictivas que pueden integrarse en estrategias preventivas, el proyecto impulsa mejoras en la calidad de vida y bienestar psicosocial de estas comunidades.

Asimismo, el proyecto se alinea con el ODS 10: Reducción de las desigualdades, al focalizarse en una minoría socialmente desfavorecida, integrando enfoques inclusivos que buscan cerrar brechas en el acceso a la innovación tecnológica y a los beneficios de la investigación en salud. De igual forma, se promueve la equidad en el uso de herramientas digitales avanzadas para apoyar decisiones clínicas, sin comprometer la privacidad ni vulnerar derechos fundamentales.

Finalmente, el uso del aprendizaje federado también se relaciona con el ODS 9: Industria, innovación e infraestructura, al incorporar tecnologías emergentes en el campo de la inteligencia artificial y garantizar un desarrollo tecnológico ético, sostenible y centrado en las personas. El proyecto, por tanto, no solo representa un avance metodológico en la

aplicación de modelos predictivos en salud, sino también un aporte tangible a los objetivos globales de desarrollo sostenible desde una perspectiva local y sensible a la desigualdad.

ANEXO II: CÓDIGO UTILIZADO

El código utilizado se puede encontrar en el repositorio: Cuadrado, Á. (2025). *TFG* (Versión de 3 de julio de 2025) [Código fuente]. GitHub. https://github.com/angelcuadrado/TFG