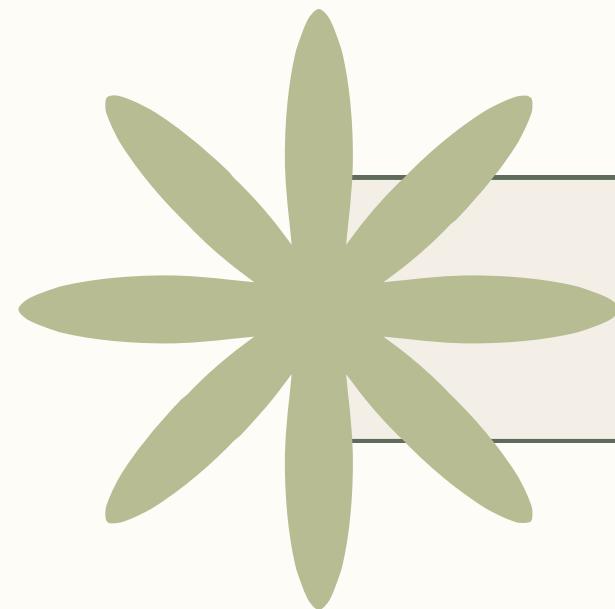


DESARROLLO Y COMPARACIÓN DE MODELOS DE APRENDIZAJE AUTOMÁTICO para la predicción del ciclo de vida útil restante (RUL) en motores turbofán: integración de Graph Neural Networks (GNN) y XGBoost.

Sara Illatopa Marín | Jose Manuel López Arias
Julio César Posada Torres | David Alvarez Grisales



ÍNDICE

01 Introducción

02 Desarrollo Metodológico

03 EDA y Preparación de Datos

04 Selección de Modelos e Ingeniería de Características

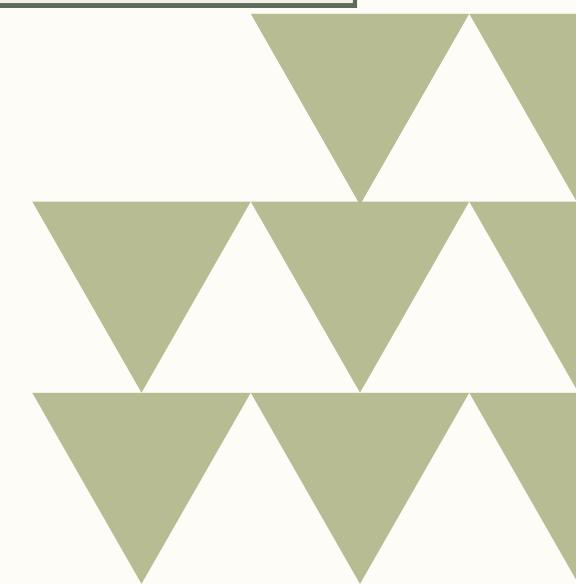
05 Marco para la Evaluación Cuantitativa

06 Configuración de modelos para comparación

07 Entrenamiento y evaluación

08 Análisis de Resultados

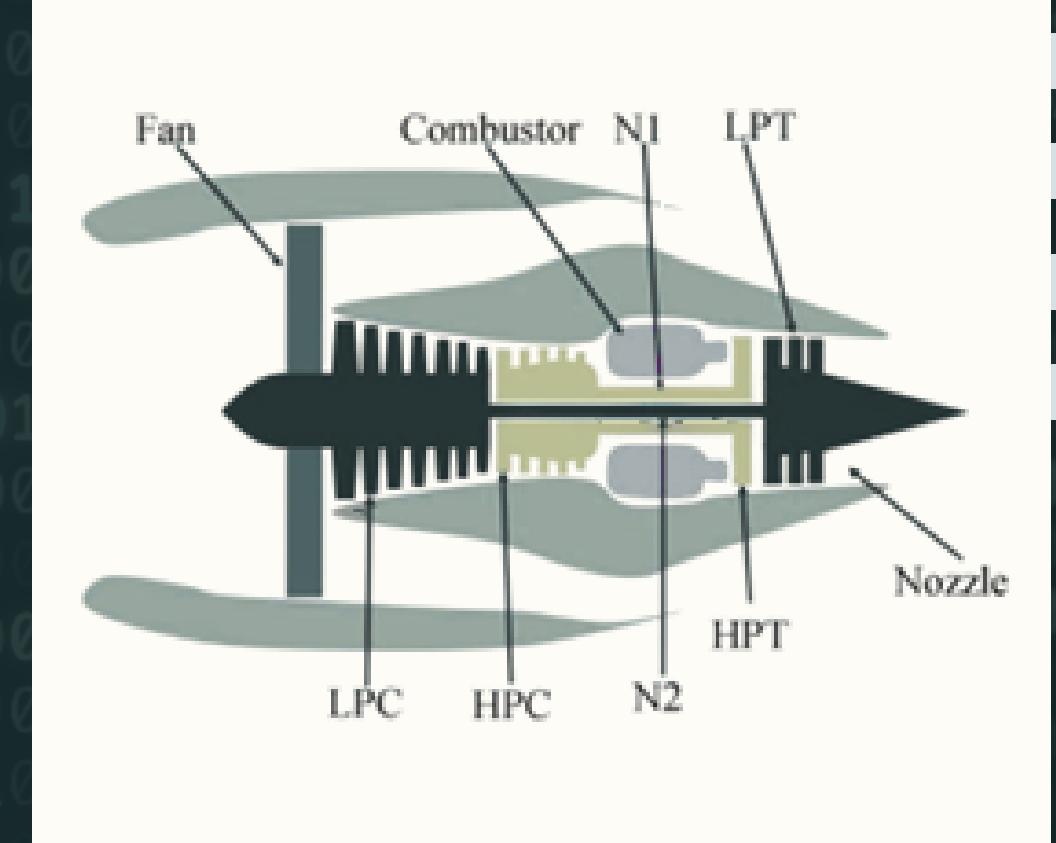
09 Conclusiones



El mantenimiento predictivo **busca anticipar fallas** usando modelos estadísticos y de machine learning.

El dataset NASA C-MAPSS es el estándar de la industria **para evaluar modelos de estimación del RUL** en motores turbofán.

La estimación del RUL es un **problema de regresión en series temporales**.



Flugra: Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation



Pregunta central: Estimar con precisión el RUL de un motor turbofán usando solo datos de sensores.

Dataset utilizado: NASA C-MAPSS, estándar para simulación y pronóstico de motores a reacción.

Características del dataset

Fuente: NASA Ames Research Center (PCoE).

Propósito: Simular degradación progresiva para validar modelos de pronóstico.

Subconjuntos (FD001–FD004): Varían en condiciones operativas y modos de falla.

Proyecto enfocado en FD001:

1 condición operativa.

1 modo de falla (degradación del compresor).

Estructura de los datos:

26 columnas: ID, ciclo, 3 configuraciones operativas, 21 sensores.

Archivos clave:

Train: 100 motores con trayectoria completa hasta falla.

Test: 100 motores con trayectoria truncada.

RUL: Ground truth para evaluación.

EDA COMO PASO ESENCIAL:

Identifica patrones, anomalías y relevancia de las variables.

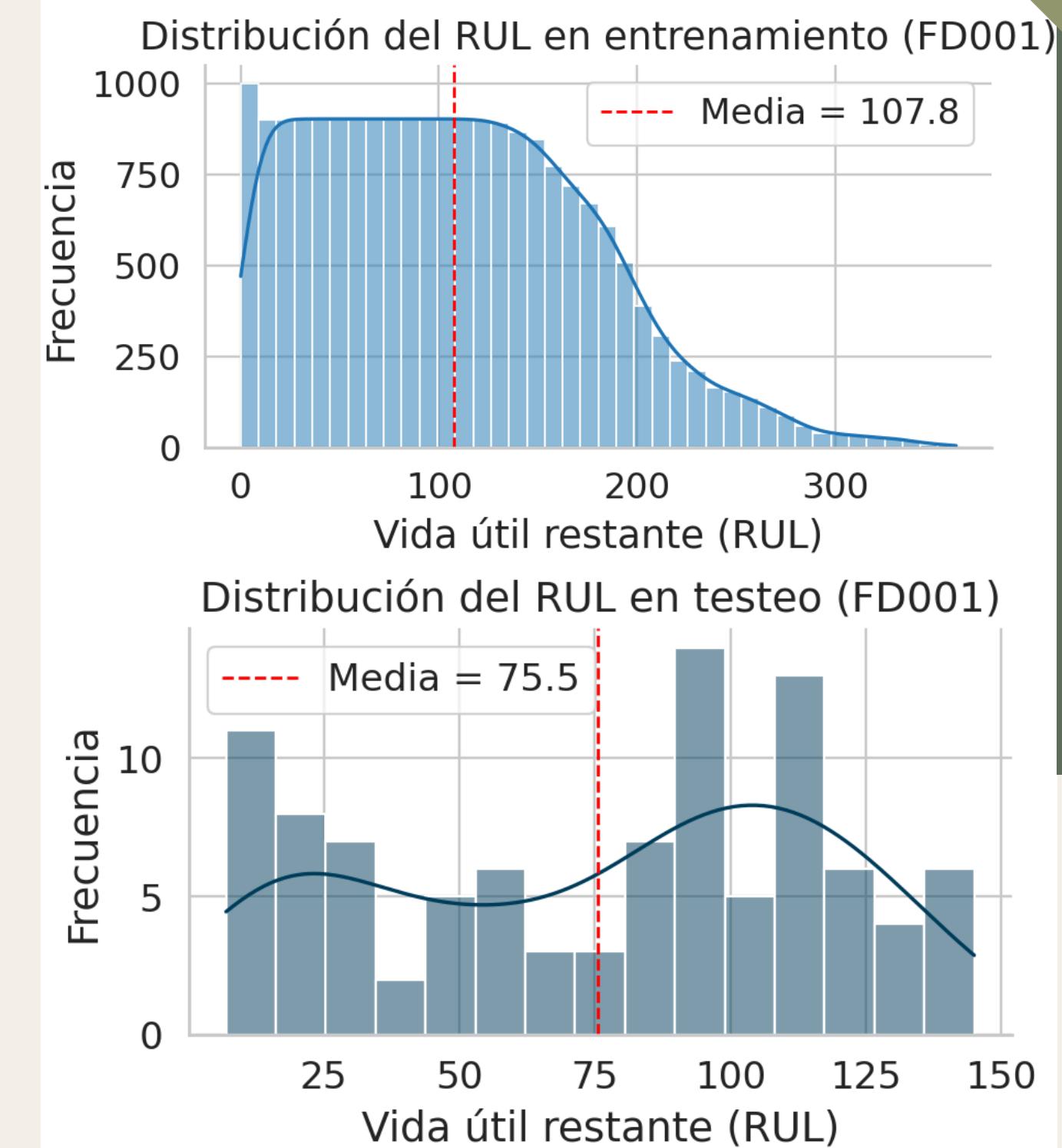
Sensores descartados: Algunos tenían varianza nula o muy baja → no aportaban información.

Variables de configuración (settings 1, 2, 3):

Baja importancia predictiva en modelos iniciales.

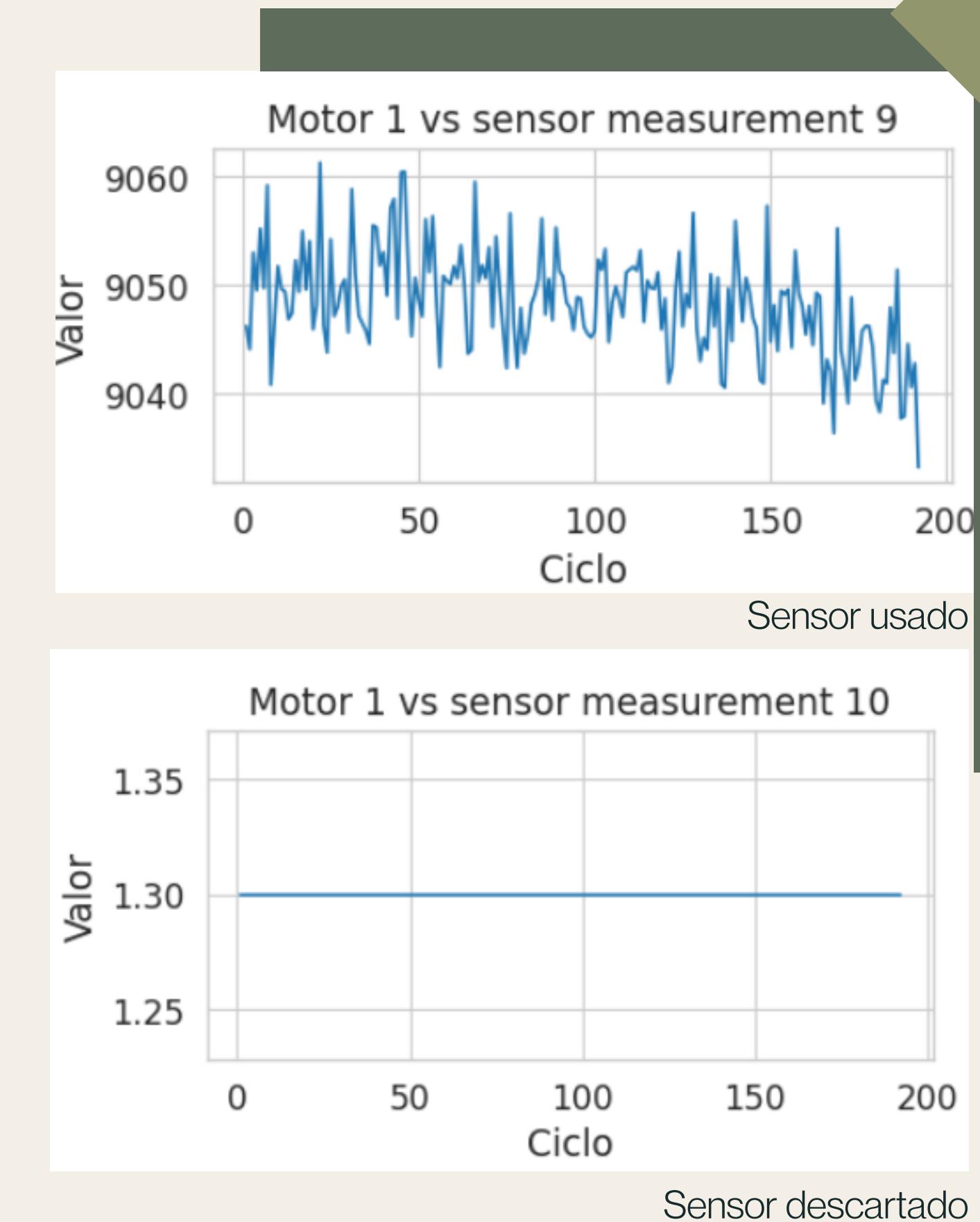
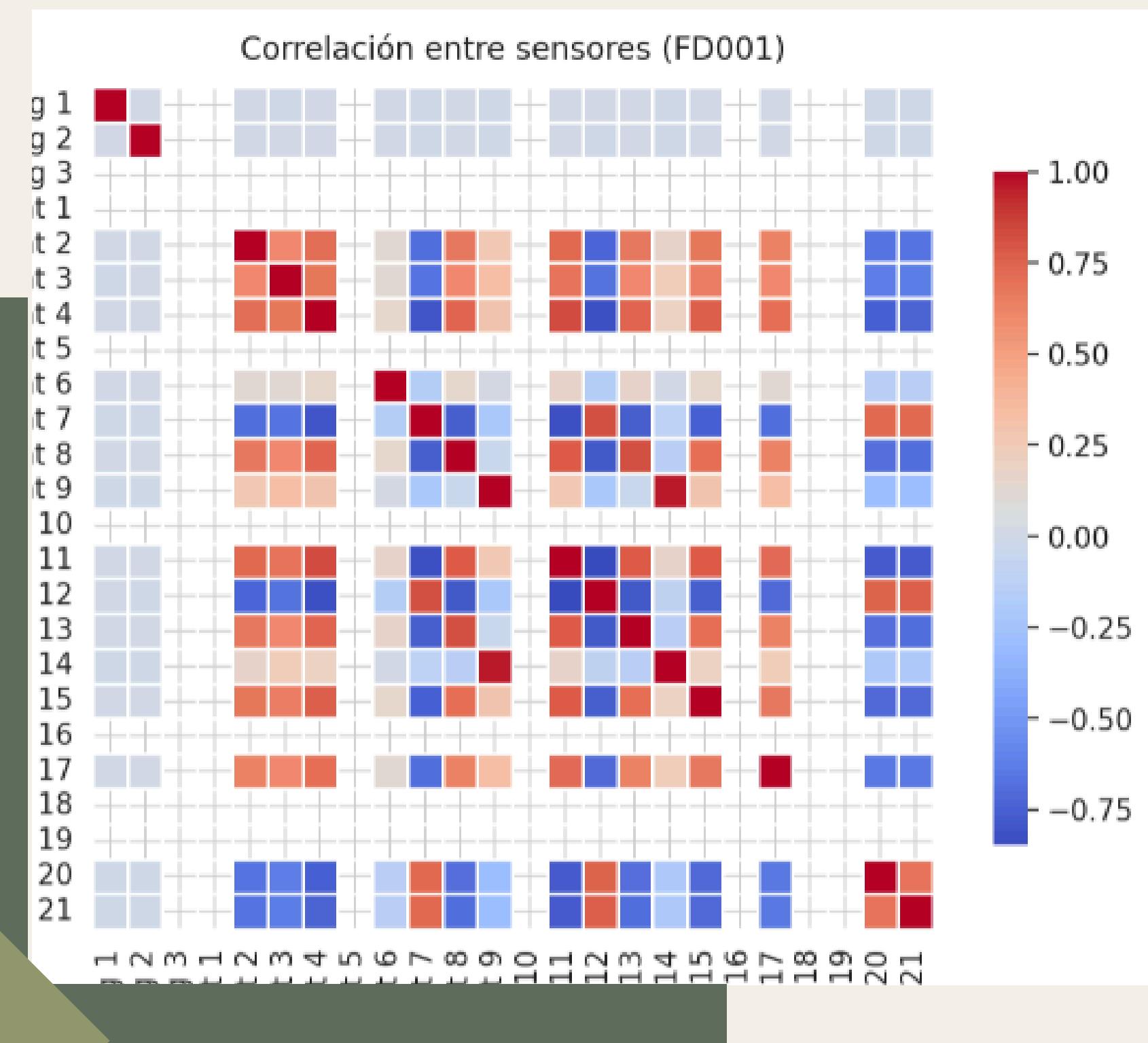
LassoCV encontró coeficientes nulos en settings 1 y 3.

Razón: FD001 trabaja con una única condición operativa → no aportan valor.



Ejemplo hallado durante el EDA

EDA y Preparación de Datos

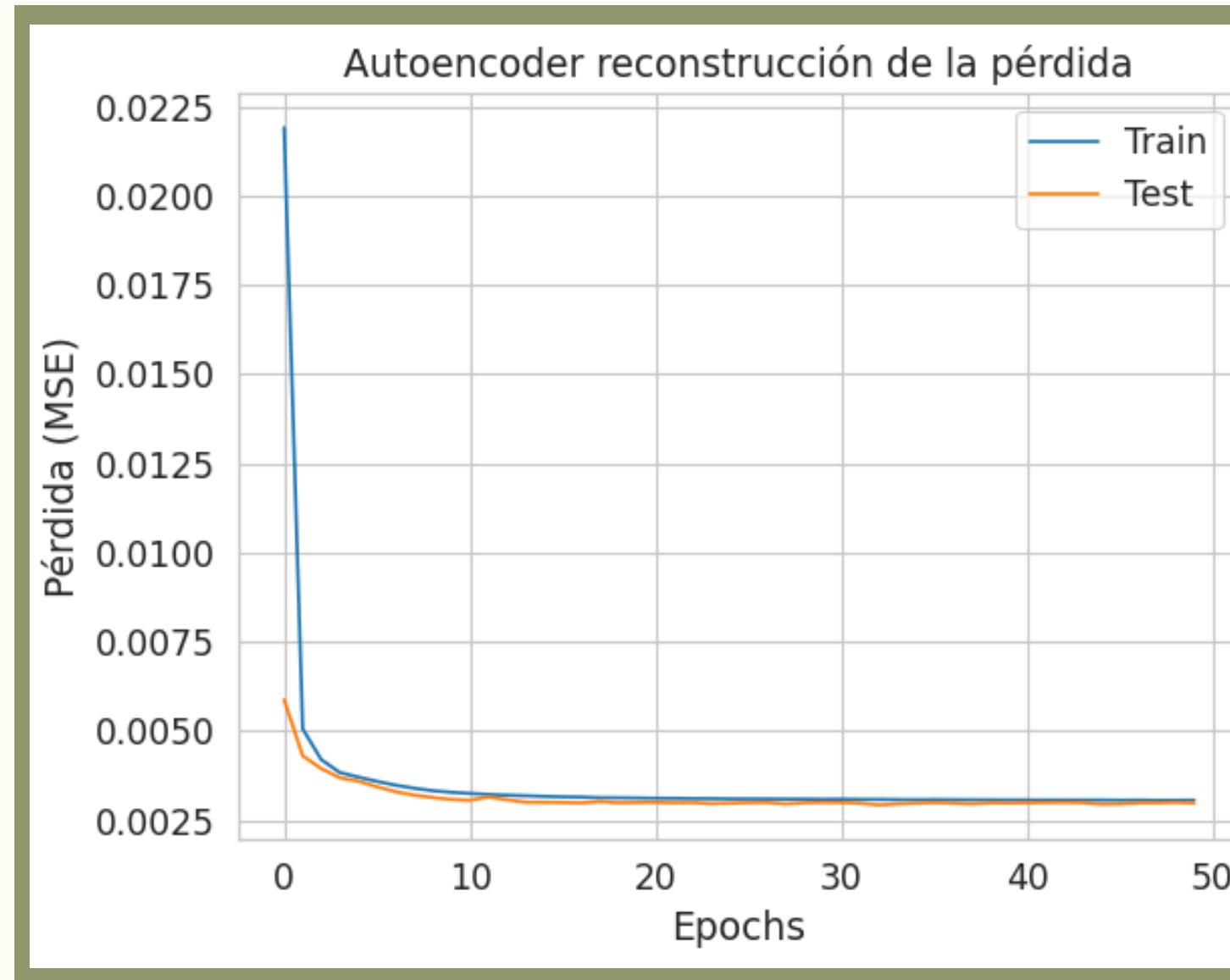




ESTRATEGIA GENERAL

Comparación entre un modelo clásico (XGBoost) y un enfoque basado en grafos (GNN).

Evaluación de importancia de variables usando métodos complementarios: XGBoost, LassoCV y Autoencoder.

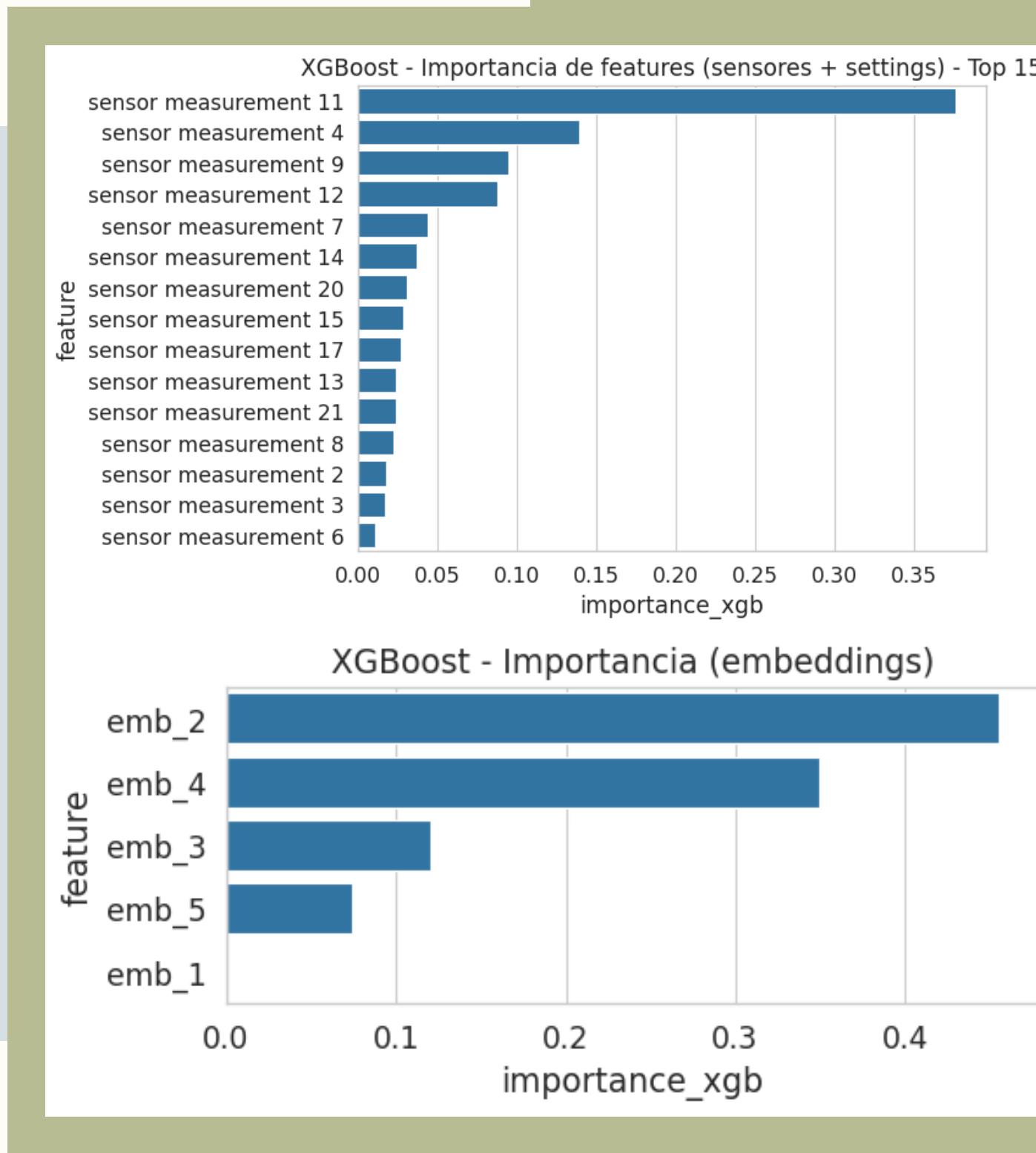


Técnicas de ingeniería de características usadas

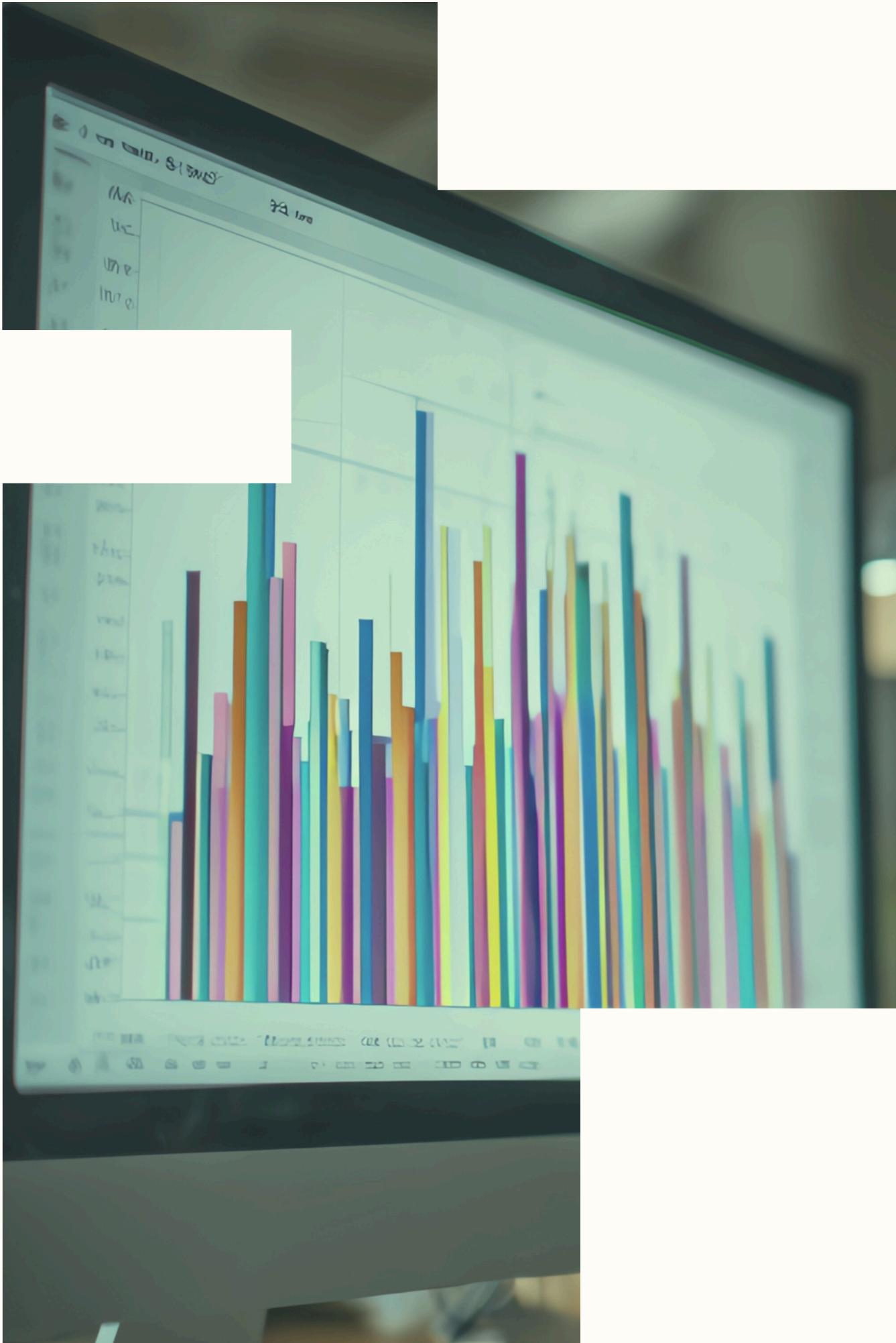
- **Importancia de características (XGBoost):** mide contribución de cada sensor al RUL.
- **LassoCV:** selecciona variables realmente informativas (coeficientes de irrelevantes $\rightarrow 0$).
- **Autoencoder:** reduce dimensionalidad filtrando ruido y extrayendo señales latentes relevantes.

Hallazgos clave

- Embeddings mejoraron mucho: XGBoost + embeddings bajó el error de forma significativa vs baseline.
- Mejor modelo: XGBoost optimizado: Obtuvo el RMSE y MAE más bajos de todo el estudio.
- GNN no superó a XGBoost: Rendimiento más alto en error incluso tras optimización.
- Menos sensores = más estabilidad: La depuración de features no afectó y mejoró el entrenamiento.
- Distribución de RUL consistente: Train amplio y decreciente; test concentrado en ciclos finales.



Marco para la Evaluación Cuantitativa



Se definió un conjunto de características limpio y validado para comparar los modelos.

La evaluación final se centrará en medir el rendimiento de XGBoost y GNN usando métricas de regresión.

La métrica principal será el RMSE, para determinar cuál modelo predice el RUL con mayor precisión.

El objetivo es verificar si la GNN ofrece una mejora significativa frente al sólido desempeño de XGBoost en este problema.

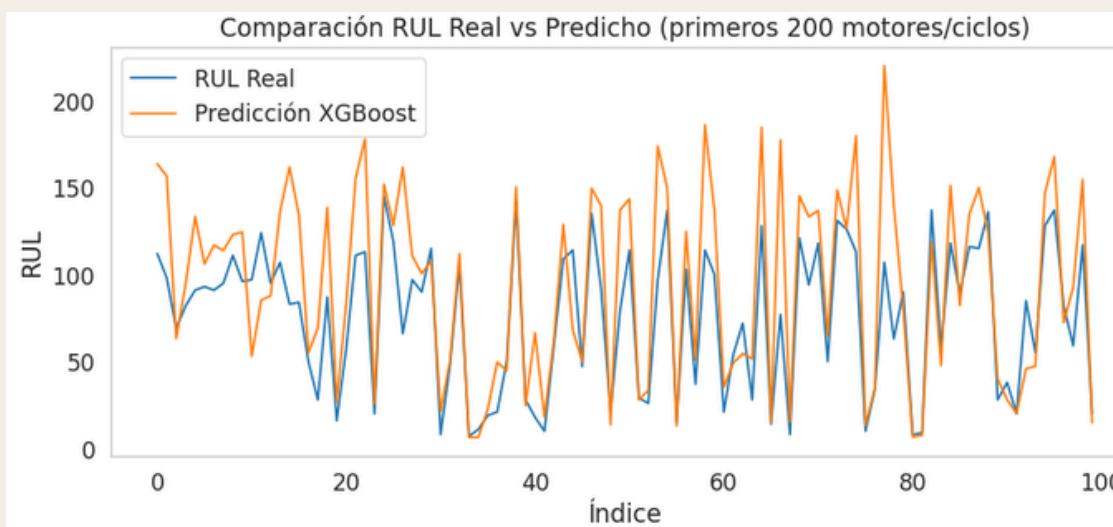
Configuración de modelos para comparación

Modelos definidos para evaluación



XGBoost Baseline

Sin selección de características.

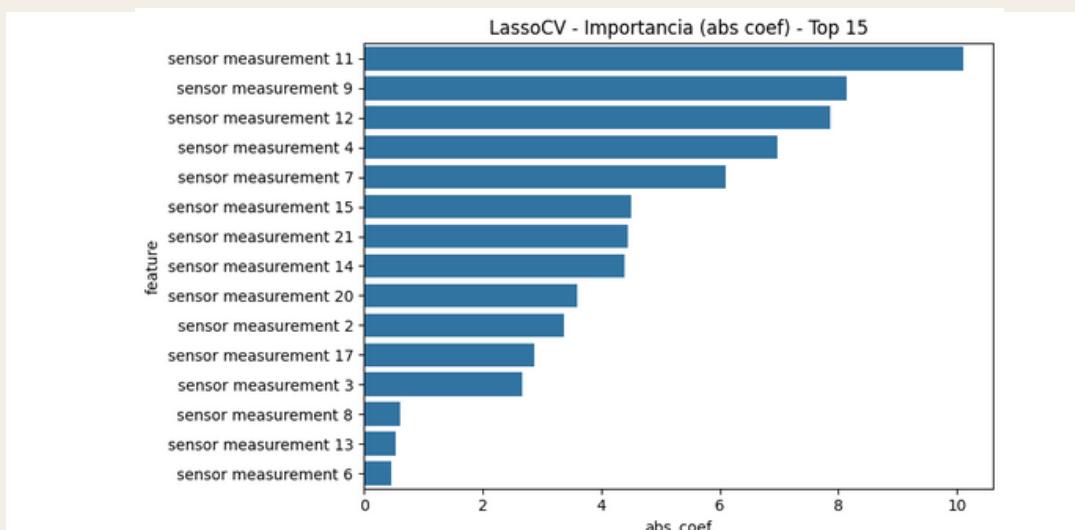


RESULTADOS XGBOOST
RMSE: 34.6716
MAE : 24.6632



XGBoost + Lasso

Usa solo los sensores más relevantes (11, 4, 9, 12).

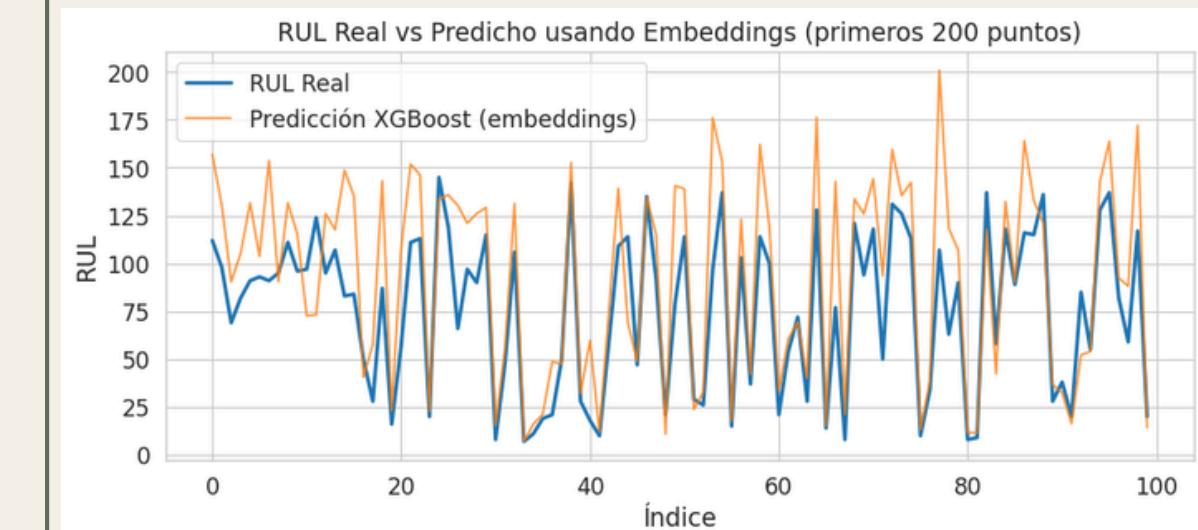


XGBoost (sensores)
RMSE: 35.0221
MAE: 24.7929



XGBoost + Embeddings

Usa el espacio latente comprimido.



XGBoost (embeddings)
RMSE: 30.6867
MAE: 23.1262

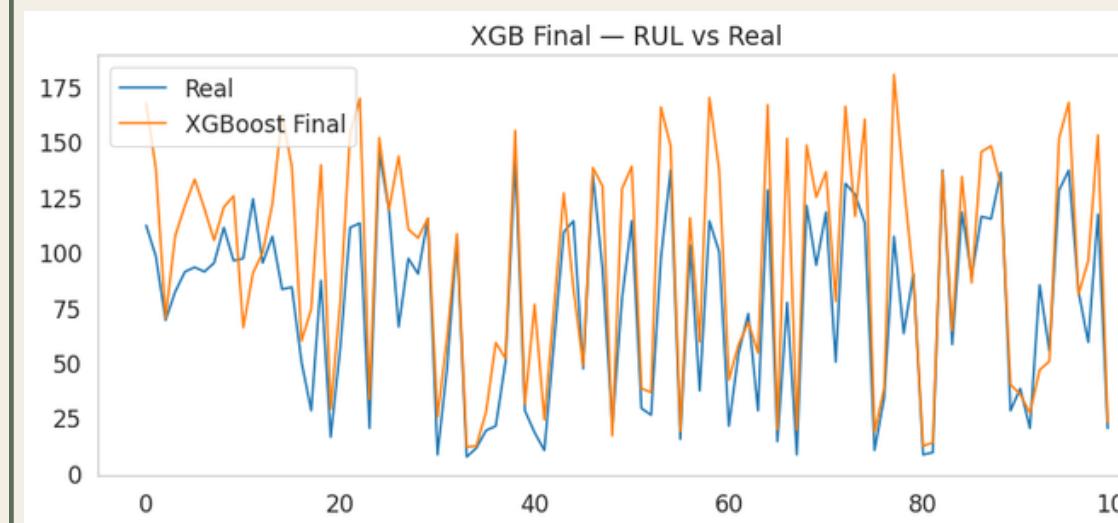
Configuración de modelos para comparación

Modelos definidos para evaluación



XGBoost optimizado

Ajuste exhaustivo de hiperparámetros.

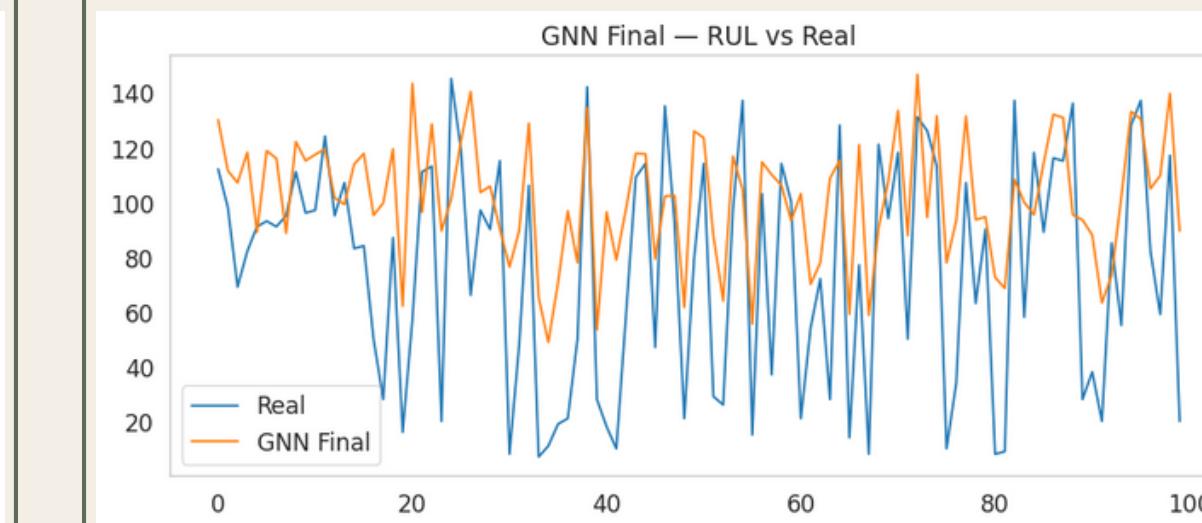


XGBOOST (OPTIMIZADO)
RMSE: 30.569801378936962
MAE : 22.798179626464844



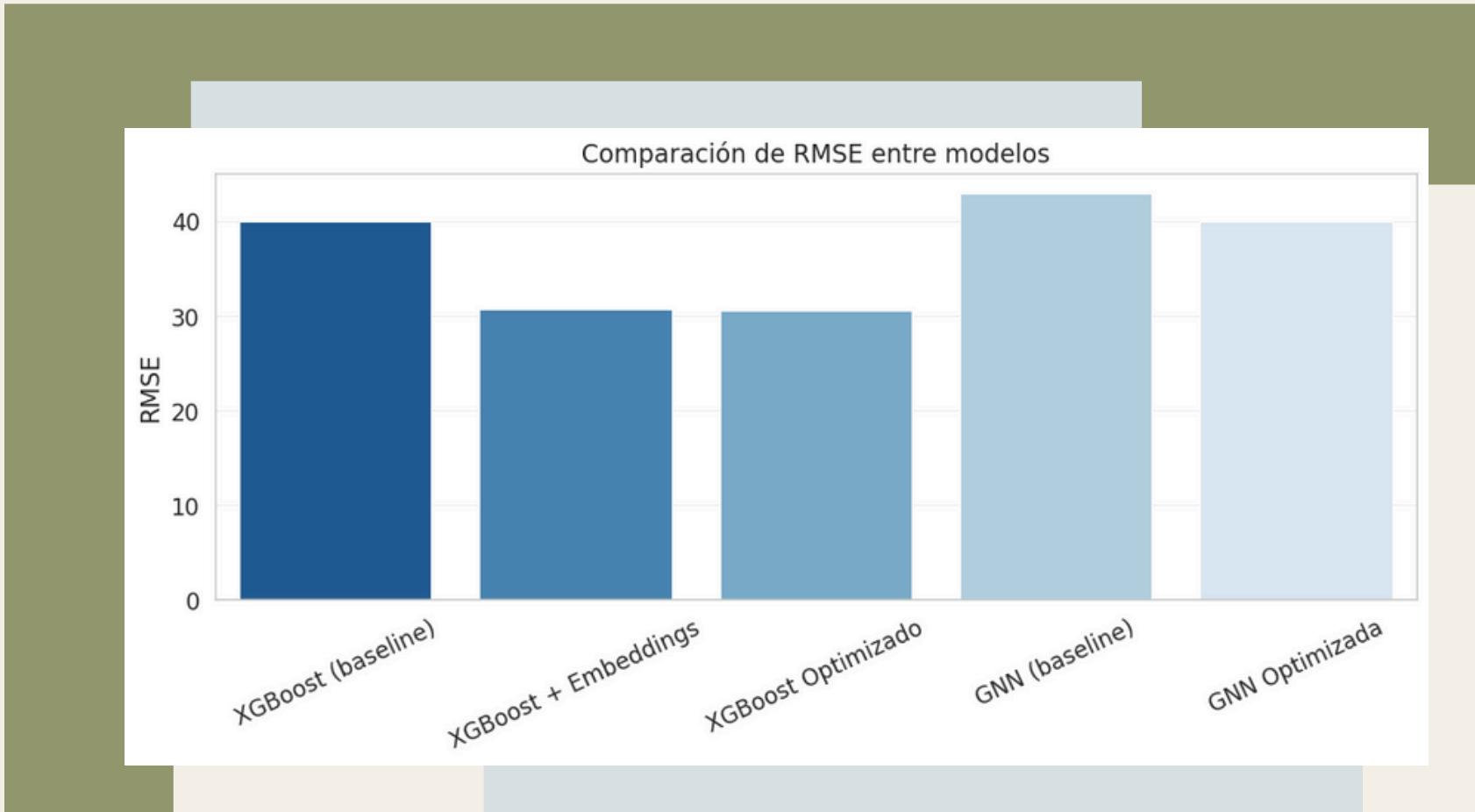
GNN optimizada

Modela relaciones entre sensores como grafos; principal aporte innovador del proyecto.



GNN FINAL (OPTIMIZADA)
RMSE: 40.003159970397185
MAE : 33.17811584472656

Entrenamiento y evaluación

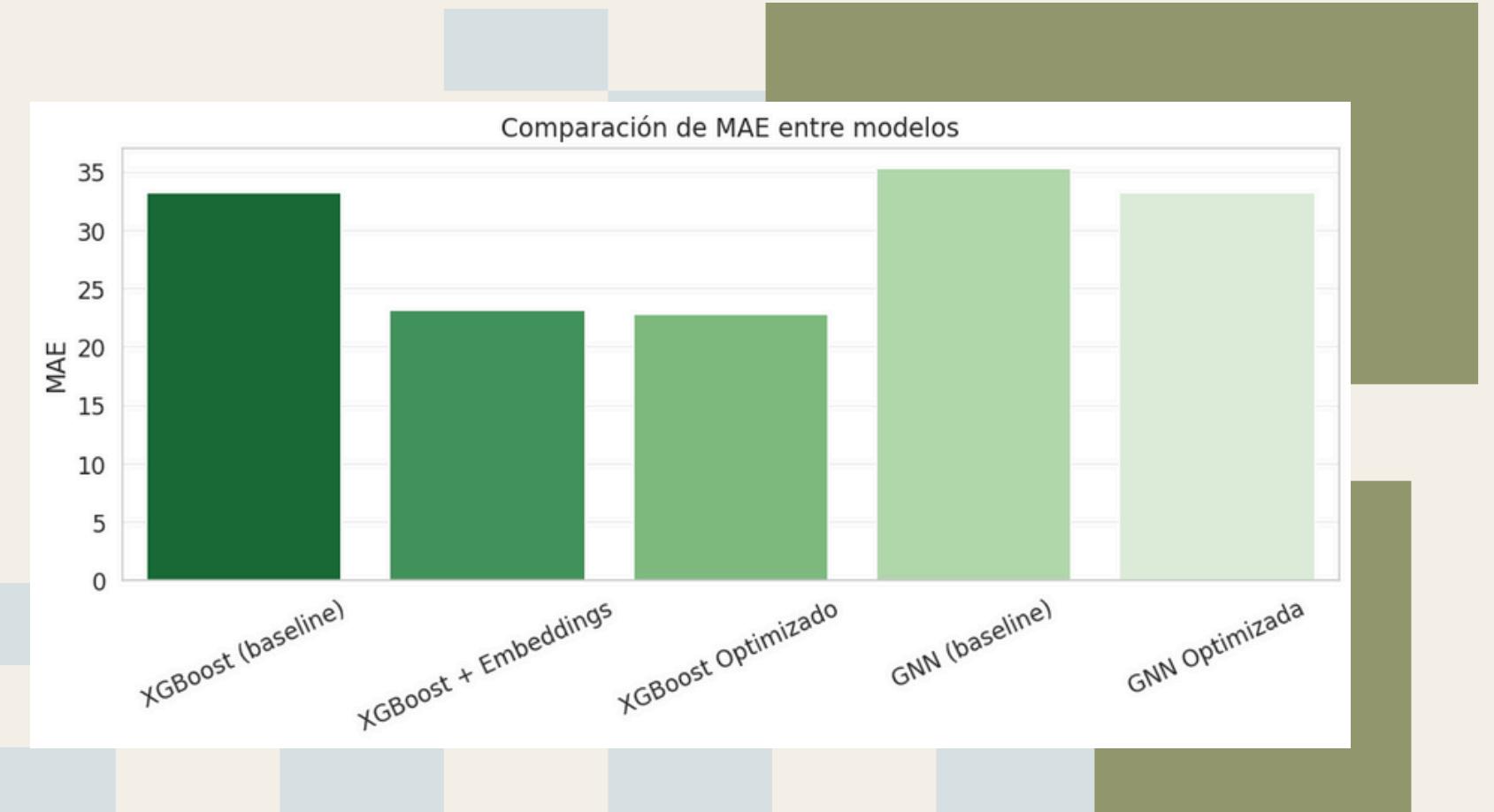


Las predicciones se compararon con los valores reales de RUL_FD001.txt para medir la precisión.

Con estos resultados, se realizó el análisis comparativo entre modelos para determinar cuál ofrece el mejor desempeño.

Los modelos (XGBoost y GNN) fueron entrenados con train_FD001.txt, aprendiendo a mapear lecturas de sensores al RUL.

La evaluación se hizo con test_FD001.txt, usando las trayectorias truncadas para generar predicciones del RUL final de cada motor.



Análisis de Resultados

01

Se identificaron exitosamente las variables más influyentes para predecir el RUL.

02

Tanto XGBoost como LassoCV coincidieron en señalar a los sensores 11, 4, 9 y 12 como los predictores más relevantes.

03

La consistencia entre ambos métodos confirma que estos sensores capturan el proceso real de degradación del compresor de alta presión en el subconjunto FD001.

04

Este hallazgo brinda alta confianza en la validez física y estadística de los modelos.

Conclusiones

- Los embeddings del autoencoder mejoraron significativamente el desempeño del XGBoost frente al modelo base.
- La optimización del XGBoost logró la menor RMSE y MAE, mostrando que el modelo clásico sigue siendo altamente competitivo.
- Las GNN no superaron a XGBoost en este escenario, indicando que requieren más ajuste o información temporal para aportar ventaja
- La optimización de hiperparámetros fue clave para el modelo ganador, mostrando que XGBoost responde muy bien a ajustes en profundidad de árboles, tasa de aprendizaje y regularización.

Conclusiones

Modelo

- Incorporar secuencias usando modelos temporales (LSTM/Transformers) para capturar degradación dinámica.
- Agregar sensores derivados (tendencias, tasas de cambio) para enriquecer las features.
- Validación cruzada por unidad para evaluar mejor la generalización entre motores.

Alcance

El estudio se limitó al subconjunto FD001.

No se evaluó el rendimiento bajo múltiples condiciones operativas o modos de falla.

Proyecto integrador

2025



**LA GANANCIA REAL VINO DE
MEJORES FEATURES, NO DE
MAYOR COMPLEJIDAD**

Muchas gracias.

Referencias

- Babu, G. S., Zhao, P., & Li, X. (2016). Deep convolutional neural network based regression approach for estimation of remaining useful life. In Database systems for advanced applications (pp. 214–228). Springer.
https://doi.org/10.1007/978-3-319-32025-0_17
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
<https://doi.org/10.1023/A:1010933404324>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
<https://doi.org/10.1162/neco.1997.9.8.1735>
- Li, X., Ding, Q., & Sun, J. Q. (2019). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*, 172, 1–11. <https://doi.org/10.1016/j.ress.2017.11.021>
- NASA Ames Research Center. (2008–2024). C-MAPSS jet engine simulated data [Dataset]. NASA Open Data. <https://data.nasa.gov/dataset/cmapss-jet-engine-simulated-data>
- Saxena, A., & Goebel, K. (2008). Turbofan engine degradation simulation data set [Dataset]. NASA Ames Prognostics Data Repository. NASA Ames Research Center.
<https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>
- Zheng, S., Ristovski, K., Farahat, A., & Gupta, C. (2017). Long short-term memory network for remaining useful life estimation. In 2017 IEEE International Conference on Prognostics and Health Management (ICPHM) (pp. 88–95). IEEE. <https://doi.org/10.1109/ICPHM.2017.7998311>
- D. Frederick, J. DeCastro, and J. Litt, "User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C MAPSS)," NASA/ARL, Technical Manual TM2007-215026, 2007.
- OpenAI. (2025). ChatGPT (GPT-5.1) [Modelo de lenguaje grande]. <https://chat.openai.com/>
- Google. (2025). Gemini [Modelo de lenguaje grande]. <https://gemini.google.com/>