Predicción de la vida útil remanente de la batería de un vehículo de movilidad sostenible

Daniel Fernando Acosta González

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesor
Olga Cecilia Úsuga Manco, Doctor (PhD)

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Antioquia, Colombia
2023

Cita	(Acosta González, 2023)
Referencia **Estilo APA 7 (2020)**	Acosta González, D. (2023). *Predicción de la vida útil remanente de la batería de vehículos de movilidad sostenible*. Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia.

Especialización en Analítica y Ciencia de Datos, Cohorte IV.

Centro de Investigación Ambientales y de Ingeniería (CIA).

UNIVERSIDAD DE ANTIOQUIA
Vicerrectoría de Docencia

Sistema de Bibliotecas

Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: http://bibliotecadigital.udea.edu.co

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes.
Decano: Julio Cesar Saldarriaga Molina
Jefe departamento: Diego José Luis Botia Valderrama

Dedicatoria

A mis padres y seres queridos.
Amar la meta tanto como el camino. Amar los logros tanto como el esfuerzo propio y el de quienes nos rodean.

Agradecimientos

A mi familia y mi novia por estar siempre ahí, tener la paciencia y la empatía en los momentos de alta carga académica y laboral. A mis compañeros de estudio por su valiosa presencia. A mis profesores y asesora por su conocimiento. A la Facultad de Ingeniería por su formación de calidad.

Tabla de contenido

Lista de tablas

Lista de figuras

Siglas, acrónimos y abreviaturas

ABR	AdaBoost Regressor
CV	Cross-validation
GBR	Gradient Boost Regressor
MSE	Mean squared error
MAE	Mean absolute error
MAPE	Mean absolute percentage error
RMSE	Root mean squared error
SVR	Support Vector Regressor
VE	Vehículos eléctricos

Resumen

Este estudio se enfoca en la predicción de la vida útil remanente (RUL, por sus siglas en inglés) de baterías de ion litio en cierto tipo de bicicletas eléctricas utilizando modelos de machine learning. Se emplearon tres modelos distintos: Gradient Boost Regressor, Adaboost Regressor y Support Vector Regressor, para predecir la variable RUL a partir de datos experimentales recopilados durante un año en vías de Medellín, Colombia. Se incluyeron variables ambientales y descriptivas del sistema elétrico. Inicialmente, se realizó una exploración gráfica de los hiperparámetros utilizando un conjunto de prueba del 70%. El 30% restante se dividió en un 20% para un segundo conjunto de prueba (Test 1) y un 10% para un tercer conjunto (Test 2). Se utilizó el conjunto de entrenamiento y Test 1 para realizar un Grid Search CV para cada modelo y seleccionar los mejores hiperparámetros basados en la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R2). Los modelos con el mejor desempeño se sometieron a una comparación adicional utilizando cross validation, resultando en la elección del Gradient Boost Regressor con tasa de aprendizaje de 0.2, profundidad máxima de 7 y número de estimadores de 220 como el modelo óptimo. Este modelo final fue evaluado utilizando el conjunto de datos Test 2, y se obtuvo las métricas de evaluación, como RMSE (6,14), MSE (37,81), MAE (23,57), MAPE (0,09) y R2 (0,99). https://github.com/IngDanielAcosta/MonografiaEspecializacionACD.

Palabras clave: Inteligencia artificial, vida útil remanente, baterías de ion litio, vehículos eléctricos

Abstract

This study focuses on predicting the Remaining Useful Life (RUL) of lithium-ion batteries in a certain type of electric bicycles using machine learning models. Three different models were employed: Gradient Boost Regressor, Adaboost Regressor, and Support Vector Regressor, to forecast the RUL based on experimental data collected over a year on roads in Medellín, Colombia. Environmental variables and descriptive features of the electrical system were included. Initially, a graphical exploration of hyperparameters was conducted using a 70% training dataset. The remaining 30% was divided into a 20% Test 1 set and a 10% Test 2 set. The training set and Test 1 were used for Grid Search CV for each model, selecting the best hyperparameters based on Root Mean Squared Error (RMSE) and Coefficient of Determination (R2). The models with the best performance underwent additional comparison using cross-validation, resulting in the selection of the Gradient Boost Regressor with a learning rate of 0.2, maximum depth of 7, and 220 estimators as the optimal model. This final model was evaluated using the Test 2 dataset, obtaining evaluation metrics such as RMSE (6.14), MSE (37.81), MAE (23.57), MAPE (0.09), and R2 (0.99). https://github.com/IngDanielAcosta/MonografiaEspecializacionACD.

keywords: machine learning, remaining useful life, Lithium-ion batteries, electric vehicles.

INTRODUCCIÓN

A medida que el enfoque global se centra en las discusiones sobre el impacto ambiental de las actividades humanas, se vuelve esencial explorar alternativas al uso de recursos energéticos no renovables y promover la adopción de energías sostenibles. Un ejemplo de esto es la incorporación de vehículos eléctricos (VE) (Ding et al., 2017). En este contexto, las autoridades gubernamentales de países desarrollados están impulsando estrategias que abarcan desde la reducción de impuestos hasta subsidios para la compra de VE, así como la expansión de la infraestructura de carga(Sanguesa et al., 2021). Por otro lado, es importante abordar las posibles repercusiones ambientales derivadas de la producción de vehículos eléctricos, especialmente en lo que respecta a las baterías. Esto requiere estrategias que mejoren la posibilidad de reciclaje y reutilización de estas baterías, junto con desarrollos tecnológicos enfocados en el monitoreo de su vida útil (Harper et al., 2019).

Las técnicas de machine learning existentes ofrecen un amplio conjunto de herramientas para predecir diversos fenómenos. En este contexto, proponemos modelos de regresión de machine learning para estimar el estado de vida útil de la batería de una bicicleta eléctrica, conocido como Remaining Useful Life (RUL). Esto implica entender las relaciones entre los factores relacionados con el desgaste de la batería o conocer las variables de contexto eléctrico ligadas al funcionamiento de la misma. Para este caso particular se tomaron las variables: voltaje del motor, voltaje de la batería, temperatura del motor, temperatura del controlador del motor, temperatura central de la batería, distancia recorrida, velocidad promedio, potencia en el motor, corriente en el motor, humedad ambiente, temperatura ambiente, corriente en batería y potencia en batería.

Este informe presenta la estimación de la vida útil restante de la batería de una bicicleta eléctrica basada en el análisis de dos baterías y las variables mencionadas, en una vía del transporte público de la ciudad de Medellín, Colombia. Para llevar a cabo esta predicción, se utilizó, ajustó y comparó tres modelos de regresión, incluyendo: Gradient Boosting para regresión, AdaBoost Regressor y Support vector regressor.

1. Descripción del problema

La vida útil de las baterías es un factor crucial en la confiabilidad de vehículos eléctricos. Se han implementado diversas metodologías para estimar la vida útil partiendo de aproximaciones de distintos indicadores. Uno de ellos es la vida útil remanente que se explora desde diferentes enfoques y su fin es proporcionar información sobre el desempeño de las baterías.

1.1. Problema de negocio

El grupo ALIADO de la Universidad de Antioquia desarrolla un proyecto de investigación para identificar patrones de desgaste y estimar la vida útil de las baterías de un tipo de bicicletas eléctricas, sometidas a pruebas de desempeño en un entorno de uso real. Este contexto implicó la experimentación en vías de la ciudad de Medellín.

A través de una revisión de literatura previa para la definición de la metodología de abordaje del problema y la experimentación con dos vehículos, se realizó la recolección de información de señales de sensores ubicados en la bicicleta, además de variables relacionadas con el entorno. Se incluyen variables relativas al sistema de la batería, y se determina la variable RUL, calculada a partir del tiempo hasta la falla (considerando falla al descargue total de la batería), como el elemento a predecir en función del voltaje, corriente y temperatura.

1.2. Aproximación desde la analítica de datos

Se desarrollaron e implementaron modelos predictivos de regresión cuyo objetivo fue el de estimar el tiempo hasta la próxima descarga representado en el RUL como variable respuesta. Las posibles variables del sistema y del entorno, tomadas en la experimentación, se establecieron como como variables de entrada de los modelos. Dada la naturaleza numérica del tiempo hasta la descarga y la existencia de la información etiquetada, se opta por la exploración y aplicación de modelos de regresión de aprendizaje supervisado. Si bien una aproximación basada en modelos de series temporales podría resultar en una aproximación adecuada, se optó por esta estrategia a fin de realizar comparaciones con los resultados de un proyecto previo apoyado en Deep Learning con el mismo fin, cuyos resultados no se incluyen en este trabajo.

1.3. Origen de los datos

Estudiantes asociados al Grupo ALIADO de la Universidad de Antioquia recolectaron información de 65 ciclos de carga y descarga en dos baterías de cierto tipo de bicicleta eléctrica, incluyendo variables del entorno como la temperatura ambiente y otras internas del sistema como el voltaje, la corriente y la temperatura de la batería y puntos del motor eléctrico. Los vehículos de prueba fueron suministrados por una compañía interesada en la estimación de la vida útil de los elementos imprescindibles en el uso de los vehículos.

1.4. Métricas de desempeño

Dada la naturaleza numérica de las cifras involucradas, resulta imperativo adoptar una métrica robusta que garantice una valoración cuantitativa equitativa, independientemente de si la estimación tiende a superar o subestimar el valor real. Además, es crucial contar con una métrica que brinde una visión transparente de la disparidad entre la estimación y el valor real experimental.

Considerando estos elementos, se decidió emplear la Raíz Cuadrada de la Media de los Errores Cuadráticos (RMSE, por sus siglas en inglés), cuyo análisis detallado se presenta en secciones posteriores. Sin embargo, como parte de un enfoque comprehensivo, se llevaron a cabo pasos adicionales que incluyeron la evaluación del R2 con el propósito de cuantificar la variabilidad explicada a partir de los modelos propuestos y otras métricas de desempeño como MSE, MAE y MAPE. Se considera importante contar con un R2 que supere 0,7 o un valor cercano a 1. Por otra parte, las demás métricas de desempeño convienen cuanto más pequeñas son. Para facilitar la visibilidad de estos resultados, se presentan las métricas de forma gráfica, negativa para los casos de métricas que se requieren pequeñas y positiva para el R2.

2. Objetivos

2.1.Objetivo general

Predecir la vida útil remanente de la batería de un modelo de bicicleta eléctrica a partir de modelos de regresión de aprendizaje supervisado.

2.2.Objetivos específicos

Preparar los datos experimentales de la vida útil remanente de la batería de un modelo de bicicleta eléctrica.

Implementar modelos de regresión de aprendizaje supervisado para la predicción de la variable objetivo.

Seleccionar el modelo de mejor desempeño en la predicción de la variable objetivo.

Establecer las características influyentes para el modelo al momento de predecir la variable objetivo.

3. Marco teórico

En esta sección se definen los antecedentes en el campo de predicción y análisis de baterías, definiciones de modelos (SVR, GBR y ABR) y métricas de desempeño (MSE, RMSE, MAE, MAPE, R2).

3.1 Antecedentes del campo

En el contexto de la predicción de la vida útil remanente en baterías, se encontraron varios aportes relacionados: Pang (Pang et al., 2023) aplicó feature separable convolution (AFSC) y convolutional long short-term memory (ConvLSTM) además de Red AFSC y ConvLSTM, comparando MAPE, RMSE y MAE. Yuan (Yuan et al., 2023) usó gradient boosting regression (GBR), support vector regression (SVR), Hist-GBR, AdaBoost, y linear regression (LR), comparando average root mean square error (ARMSE) y relative error (RE). Zhang (Zhang & Zhao, 2023) utilizó SVR y Gaussian Process Regression (GPR) comparando MAPE y RMSE. Zhao (Zhao et al., 2023) se apoyó en un modelo de conjunto basado en una estrategia de apilamiento con LR, Randon Forest Regressor (RFR), GBR y GPR. Incluso, otros autores como (Bracale et al., 2023) se apoyan en estrategias de series de tiempo, incluyendo AutoRegressive Integrated Moving Average model with eXogenous predictors, ARIMAX y estrategias de regresión como Linear Quantile Regression (LQR), Bootstrap Multiple Linear Regression (B-MLR) y Bayesian Bootstrap Multiple Linear Regression (BB-MLR). Hay otros ejemplos en contextos similares como colonias de abejas y SVR (Chen et al., 2023) y SVM (Patil et al., 2015).

3.2 Modelos

Se exploraron tres modelos de inteligencia artificial enfocados en regresión. La elección de los modelos parte de su interpretabilidad y uso frecuente en el contexto predictivo, lo que brinda documentación suficiente para conocer las implicaciones derivadas de los cambios efectuados en estos. Se exploran los modelos: Support vector regression, gradient boost regression y adaboost regression.

Support Vector Regression

La regresión de vectores de soporte (SVR, por sus siglas en inglés) proviene de las Máquinas de Soporte Vectorial (SVM), cuya formulación incorpora el principio de minimizar el riesgo estructural y se ha demostrado que supera el enfoque convencional de minimizar el riesgo empírico comúnmente utilizado en técnicas tradicionales de aprendizaje automático. Inicialmente diseñado para abordar tareas de clasificación, el SVM se ha expandido desde entonces para abarcar problemas de regresión (Roy & Chakraborty, 2023).

Consideremos un conjunto de datos $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ con $x_i \in \mathbb{R}^n$ y $y_i \in \mathbb{R}$, donde x_i es una variable de entrada correspondiente a y_i, y n es el número de muestras. Supongamos que existe un conjunto de funciones capaces de transformar un punto desde el espacio n-dimensional $\mathbb{R}^n$ al espacio de números reales $\mathbb{R}$.

$$F = \{f(\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{w}\epsilon\Lambda | f: \mathbb{R}^n \to \mathbb{R}\}$$

Aquí, Λ representa un grupo de parámetros y w es un vector de parámetros cuyo valor debe determinarse. En este contexto, la regresión implica encontrar una función f que pertenezca al conjunto F de manera que minimice el riesgo esperado. Este riesgo es una función de error que puede ser gaussiana, cuadrática, laplaciana, de módulo mínimo, robusta de Huber, entre otras(Roy & Chakraborty, 2023).

En la Regresión de Vectores de Soporte (SVR), un kernel funciona como una función matemática que transforma los datos de entrada en un espacio de mayor dimensión, permitiendo la separabilidad lineal. Esta transformación dota a SVR con la capacidad para identificar el límite de decisión óptimo, incluso cuando se enfrenta a relaciones intrincadas y no lineales entre los puntos de datos (Mathew et al., 2017). Es común encontrar tipos de kernel lineales, que estiman un hiperplano lineal; polinomiales, que, en función de un grado escogido, establecen curvaturas de un hiperplano y generan nuevas variables como combinaciones polinomiales de las previas; de Función de Base Radial (RBF), que establecen un rango radial de inclusión frecuentemente afectado por el parámetro gamma; entre otros (Rochim et al., 2021). Para los kernels RBF y

polinómicos, se puede ajustar un parámetro conocido como gamma, que determina el grado de influencia que ejerce cada punto sobre sus puntos vecinos. Un gamma bajo corresponde a puntos distantes, mientras que un gamma alto implica puntos cercanos. (Laeli et al., 2020a).

Las máquinas de vectores de soporte utilizan un parámetro de regularización, denotado como C, para encontrar un equilibrio entre la complejidad del modelo y su riesgo empírico, es decir, la cantidad de datos erróneamente definidos y la capacidad de generalización (Jordaan & Smits, s. f.). C parte generalmente de valores mayores que cero y se ajusta en función del conjunto de datos analizado. Un valor bajo de C tiende a subajustar, mientras que uno muy grande puede llevar al sobreajuste (Laeli et al., 2020b).

Gradient Boost Regressor

El Gradient Boost Regressor (GBR) es una técnica que construye progresivamente un grupo de árboles de decisión para hacer predicciones. El proceso comienza con un modelo inicial débil, típicamente un árbol de decisión poco profundo, y luego incorpora árboles adicionales en el conjunto de modelos. Cada nuevo árbol se entrena para corregir los errores de sus predecesores minimizando una función de pérdida a través del descenso de gradiente. GBR combina las predicciones de múltiples modelos débiles para generar una predicción definitiva (Ekanayake et al., 2023).

GBR consiste en una serie de pasos, empezando por la inicialización del modelo con un valor constante, así:

$$F_0(x) = \underset{\gamma}{argmin} \sum_{i=1}^{n} L(y_i, \gamma)$$

donde n es el número de muestras en y. L es la función de pérdida, usualmente una pérdida cuadrática $L = (y_i - \gamma)^2$ y, en cuyo caso el valor de γ que minimiza la sumatoria es la media de y, $\overline{y}$.

Luego, el algoritmo itera desde $m = 1$ hasta M, donde M es número de árboles construidos. Posteriormente, calcula los residuales

$$r_{im} = -\left[\frac{\partial L\left(y_i, F(x_i)\right)}{F(x_i)}\right]_{F(x)=F_{m-1}(x)} \qquad \forall\, i = 1, \cdots, n$$

Luego, el algoritmo ajusta el árbol de regresión con las características x comparadas con los residuales r creando las regiones terminales R_{jm} para $j = 1, \cdots, J_m$ y calcula el argumento mínimo, siendo m el índice del árbol y J el número correspondiente de hojas de salida.

$$\gamma_{jm} = \underset{\gamma}{argmin} \sum_{x_i \in R_{jm}} L(y_i, F_{m-1}(x_i) + \gamma)\ \forall j = 1, \cdots, J_m$$

posteriormente, se actualiza el modelo:

$$F_m(x) = F_{m-1}(x) + v \sum_{j=1}^{J_m} \gamma_{jm} 1(x \in R_{jm})$$

donde v es la tasa de aprendizaje que varía entre 0 y 1 (Carvalho & de Assis de Souza Filho, 2021). Un valor de v más pequeño implica una reducción en el efecto de un árbol adicional, pero, a su vez, reduce el riesgo de sobreajuste (Datta et al., 2022).

AdaBoost Regressor

AdaBoost regression usa habitualmente un algoritmo conocido como AdaBoost R2. Este empieza por ajustar un regresor, usualmente un árbol débil, en el conjunto de datos original. Luego, ajusta nuevos regresores del mismo tipo cambiando los pesos de los registros de acuerdo con el error de predicción haciendo que el siguiente regresor pueda concentrarse en los casos más difíciles de acertar.

Se asume un conjunto de m muestras $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ donde $y_i \in \mathbb{R}$, luego, se define un regresor débil que itera T veces. El algoritmo se inicializa con una distribución de peso $D_t(i) = \frac{1}{m}\ \forall i$ y una función de pérdida promedio $\bar{L_t} = 0$.

Mientras $\bar{L_t} < 0.5$, se llama un regresor débil con una distribución y se construye un modelo de regresión $f_t(x) \to y$. Posteriormente, se calcula la pérdida para cada muestra de entrenamiento $l_t(i) = |f_t(x_i) - y_i|$ y se calcula la función de pérdida usando tres posibles aproximaciones:

Lineal: $L_t(i) = \frac{l_t(i)}{Den_t}$.

Cuadrática: $L_t(i) = \left(\frac{l_t(i)}{Den_t}\right)^2$

Exponencial: $L_t(i) = 1 - e^{\frac{-l_t(i)}{Den_t}}$

donde $Den_t = \underset{i=1,\cdots,m}{max}(l_t(i))$

Luego, se calcula la pérdida promedio como $\bar{L_t} = \sum_{i=1}^{m} L_t(i) D_t(i)$

Posteriormente, se establece $\beta_t = \frac{\bar{L_t}}{1-\bar{L_t}}$

Se actualiza la distribución: $D_{t+1}(i) = \frac{D_t(i)\beta_t^{1-L_t(i)}}{Z_t}$ donde Z_t es el factor de normalización escogido tal que D_{t+1} sea una distribución.

Finalmente, se establece $t = t + 1$ y la hipótesis de salida:

$$f_{fin}(x) = inf\left[y \in Y: \sum_{t:f_t(x)\leq y} \log\left(\frac{1}{\beta_t}\right) \geq \frac{1}{2}\sum_t \log\left(\frac{1}{\beta_t}\right)\right]$$

En la práctica, AdaBoost regressor usualmente incorpora el estimador, el número de estimadores, la tasa de aprendizaje y el tipo de función de pérdida como parámetros del modelo (Solomatine & Shrestha, s. f.).

3.3 Métricas de desempeño

Para el presente trabajo se definen las métricas utilizadas en las diferentes etapas de evaluación de los modelos implementados.

Error cuadrático medio (MSE)

Partiendo de n muestras, para cada muestra i se calcula la diferencia entre el dato real y_i y la predicción $\hat{y}_i$. Esta se eleva al cuadrado para solventar resultados negativos y positivos, elevando la importancia del error al cuadrado (Ding et al., 2017).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Raíz del error cuadrático medio (RMSE)

A diferencia del MSE, se aplica raíz cuadrada, permitiendo llevar el resultado a las unidades de la variable y, lo que facilita su interpretación, sin afectar el rango o posición según el desempeño de los modelos probados (Hodson, 2022).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Error absoluto medio (MAE)

El error absoluto medio no eleva al cuadrado el error, manteniendo un valor relativamente más pequeño de la diferencia, pero solventando también los valores positivos y negativos. (Hodson, 2022).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Error absoluto medio porcentual (MAPE)

Es más conveniente cuando se quiere evaluar variaciones relativas más que variaciones absolutas (de Myttenaere et al., 2016).

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}|\frac{y_i - \hat{y}_i}{y_i}|$$

Coeficiente de determinación (R2)

Este valor puede ser interpretado como la proporción de la varianza en una variable independiente que es predecible a partir de las variables independientes (Dodge Yadola, 2008).

$$1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(\overline{y_i} - \hat{y}_i)^2}$$

4. Metodología

Se estableció un flujo basado en la metodología CRISP-DM (Schröer et al., 2021), adaptando las diferentes fases, habitualmente enfocadas en la implementación de los modelos de machine learning en producción, a un contexto de investigación. Con esta adaptación, se establecen

5 fases en las que el conjunto de datos experimenta transformaciones y aplicaciones con el objetivo de predecir la variable respuesta.

4.1 Comprensión del problema de investigación

En esta etapa se busca comprender el contexto de los datos para hacer un uso eficiente e inteligente de estos. Se establece un objetivo relacionado con cierta variable de salida y las posibilidades de aplicaciones que puedan resultar útiles en el contexto.

Se tomó un conjunto de datos experimental con la medición de variables relativas al sistema y el entorno de la batería de un tipo de bicicleta eléctrica. Dada la importancia de la batería en este tipo de vehículos, se estableció como objetivo estimar la vida útil remanente de la batería a partir de las variables recolectadas apoyándose en modelos de regresión de aprendizaje supervisado. Esta sección se explora con precisión más adelante.

4.2 Análisis descriptivo de los datos

En esta fase se explora la composición de la información a través de la revisión de las distintas variables y sus correspondientes relaciones. Se busca establecer proporciones de las variables categóricas, distribuciones de las variables cuantitativas, correlaciones, entre otras alternativas que aportan a la visión general del conjunto de datos y las acciones a ejecutar para la posterior limpieza. Para la exploración de las variables es posible usar métodos gráficos que proveen conclusiones sobre la composición de los datos.

4.3 Preparación de los datos

La preparación de los datos involucra todos los procesos necesarios para garantizar calidad en los datos. Para ello, se debe realizar una exploración de nulidades, registros atípicos y otros tipos de posibles errores, eliminando la menor cantidad de información posible, manteniendo la coherencia y seguimiento del proceso natural de la información. Adicionalmente, se requiere procesos de imputación para complementar el proceso de preparación. Algunos elementos requieren la revisión de expertos y otros se abordan desde una perspectiva estadística o analítica. Finalmente se establecen los conjuntos de prueba y entrenamiento a usar en las validaciones de los modelos.

4.4 Construcción de modelos de regresión

En esta etapa se define qué modelos se usan en función de la configuración de los datos y los objetivos propuestos. Para este caso específico, se requieren modelos de inteligencia artificial enfocados en regresión y que faciliten la interpretación de resultados y pesos de variables. Por ello se consideraron los modelos Gradient Boost Regressor, AdaBoost Regressor y Support Vector Machine Regressor. Estos modelos pasaron por una validación gráfica previa de los hiperparámetros y luego a través de GridSearch para encontrar las configuraciones de mejor desempeño.

Resultados y discusión: evaluación y selección de modelos

Esta etapa involucra la evaluación del desempeño de los modelos y su posterior selección, y comparación, buscando el mejor desempeño posible. En este caso específico, el desempeño de los modelos probados en GridSearch se mide en función del RMSE y R2. Posteriormente se realizará una validación cruzada con el conjunto de datos de entrenamiento y el primer conjunto de prueba para definir el modelo que presenta mejor comportamiento. A esta etapa se le añadió una validación final con el porcentaje restante de validación, llegando a la conclusión del mejor modelo revisando RMSE, MSE, MAE, MAPE y R2.

5. Comprensión del problema de investigación

Comprender el problema de investigación es fundamental para entender cómo enfocar los recursos disponibles y evaluar la factibilidad de la aplicación que se busca llevar a cabo. Con esto en cuenta, se buscó reconocer la importancia de la medida de la vida útil remanente en las baterías de ciertos vehículos eléctricos, apoyándose en metodologías de confiabilidad y trabajos con objetivos similares. El problema del RUL como medida instantánea permite abordar los datos desde una perspectiva completa y amplia, en lugar de abordar la investigación con un valor único del RUL para cada ciclo de carga. Si se recoge un valor de RUL por cada instante medido, se puede observar la incidencia instantánea de otras variables involucradas. Por otra parte, hay que reconocer que partiendo de un conjunto experimental de datos, es probable que se encuentre diversos inconvenientes de recolección de información que se espera solventar en las etapas posteriores.

En el contexto de la implementación de estrategias enfocadas en medios de transporte con energías limpias, se hace crucial garantizar un funcionamiento adecuado de los vehículos eléctricos, apoyándose en investigaciones que permitan establecer la vida útil de los componentes y determinar patrones de desgaste en ellos. Específicamente, en términos del análisis de la vida útil de las baterías, es común encontrar enfoques relacionados con indicadores que proveen conclusiones sobre la salud del sistema, la vida útil remanente, entre otros, los cuales se apoyan en técnicas de diversa índole, resumiéndose en 3 principales enfoques: basados en modelos, enfocados en datos y enfoque híbrido. En el contexto basado en datos, las estimaciones de la variable RUL, se apoyan en machine learning, aproximaciones estadísticas y procesamiento de señales (Ansari et al., 2022).

El RUL comprende diversas definiciones que emplean los datos disponibles y los enfoques de los investigadores. Para este caso en concreto, se espera pronosticar el tiempo esperado de descarga en función de las variables experimentales tomadas, considerando el RUL como el tiempo hasta la falla, donde la falla implica la descarga total de la batería. Para este fin se dispone de la toma de muestras segundo a segundo de 65 ciclos y sus respectivas variables. El RUL definido se establece bajo la siguiente ecuación:

$$RUL_{i,j} = t_{n,j} - t_{i,j} + 1 \quad \forall j$$

donde i representa cada segundo i dentro de un ciclo j y n simboliza el último segundo del ciclo j correspondiente. La variable RUL permite estimar el tiempo hasta la descarga en función del valor de las señales suministradas por un sistema instalado en el motor eléctrico y la batería de los vehículos usados para la experimentación.

6. Análisis de los datos

La etapa de análisis de datos es crucial para identificar oportunidades de aplicación, relaciones entre variables, elementos a imputar, entre otros. Se parte de la exploración de los datos originales, seguido de analítica descriptiva.

6.1 Datos originales

Se cuenta con un archivo previamente consolidado en formato csv (comma separated values) que incluye información tomada por estudiantes asociados al Grupo ALIADO de la Universidad de Antioquia, quienes recolectaron registros del seguimiento de 65 ciclos de carga y descarga de 2 baterías iguales y dos motores iguales de un modelo de bicicleta eléctrica para un total de 91.811 registros, a través del seguimiento de pruebas sobre condiciones de tráfico real en dos vías con diferentes particularidades y, por motivos climáticos, algunas pruebas se llevaron a cabo en laboratorio. Estos 65 ciclos contienen información de corriente, voltaje, potencia y temperatura en la batería y mediciones de estas variables al motor en sus 3 polos. Se recolectaron además la humedad ambiental y la temperatura ambiente. Por otra parte, se tiene información de la aceleración en 3 ejes y la duración, distancia, velocidad máxima y velocidad media durante el ciclo. También se conoce el peso del conductor. Esta información se recolectó a lo largo de un año y, teniendo en cuenta el contexto experimental, el proceso de recolección de los datos presentó algunas dificultades en la toma de muestras con los sensores o registros nulos en distintas variables, aspectos que deben solucionarse mediante el procesamiento adecuado de estos.

Se parte entonces de 91.811 registros con 33 columnas. En la **Tabla *1*** se presenta el listado de variables con su respectiva descripción.

Tabla 1

Descripción de variables iniciales

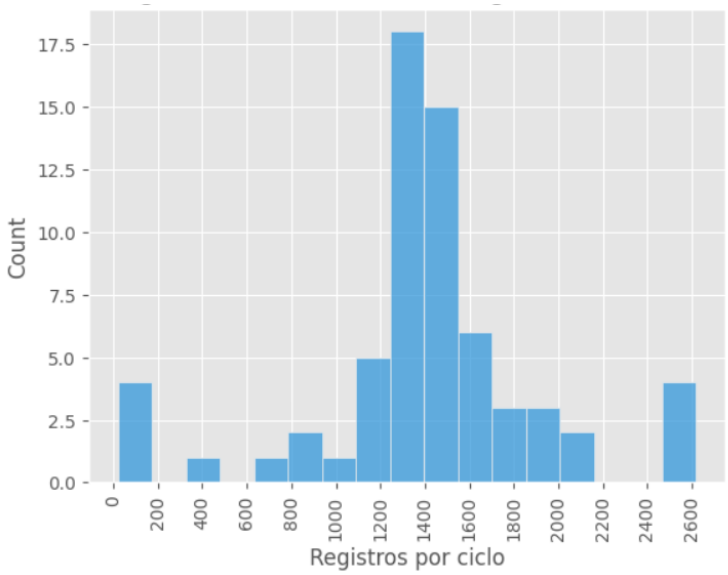
Columna	Descripción
tiempo	indicativo del segundo en que se toma información (conteo del instante)
fecha_exp	nombre y fecha del experimento. Corresponde inicialmente con el ciclo
fecha	fecha de experimento
exp	nombre del experimento
motor	motor 1 o 2
bateria	batería 1 o 2
peso_cond	peso del conductor
lugar	lugar de la prueba
duracion	duración total del ciclo
distancia	distancia recorrida
vel_max	velocidad máxima en el ciclo
vel_prom	velocidad promedio en el ciclo
ACCELERATION_X	aceleración en eje x
ACCELERATION_Y	aceleración en eje y
ACCELERATION_Z	aceleración en eje z
CURRENT_A_CALC	corriente calculada en
CURRENT_B_CALC	corriente en motor
CURRENT_C_CALC	corriente en motor en punto c
CURRENT_D_CALC	corriente en motor en punto d
POWER_A	Potencia en la batería
POWER_B	Potencia en el motor
POWER_C	Potencia en el motor en punto c
POWER_D	Potencia en motor en punto d
TEMPERATURE_A	temperatura en el motor
TEMPERATURE_B	temperatura en el controlador
TEMPERATURE_C	temperatura en el centro de la batería
TEMPERATURE_D	temperatura para el extremo de la batería
VOLTAGE_A	voltaje en batería
VOLTAGE_B	voltaje en motor
VOLTAGE_C	voltaje en punto c en el motor
VOLTAGE_D	voltaje en punto d en el motor
ENV_HUMIDITY	humedad ambiental
ENV_TEMPERATURE	temperatura ambiente

Se definió manualmente los tipos de datos de las columnas a través de un listado de variables numéricas y categóricas.

6.2 Analítica descriptiva

Cada ciclo contiene una cantidad diferente de registros, presentando la distribución que puede apreciarse en la **Figura *1***. Esto implica que ciertos ciclos contienen una cantidad muy pequeña de registros mientras que otros presentan un mayor volumen. En general, se observa una distribución que se asemeja a una normal con una media aproximada de 1400 registros por ciclo.

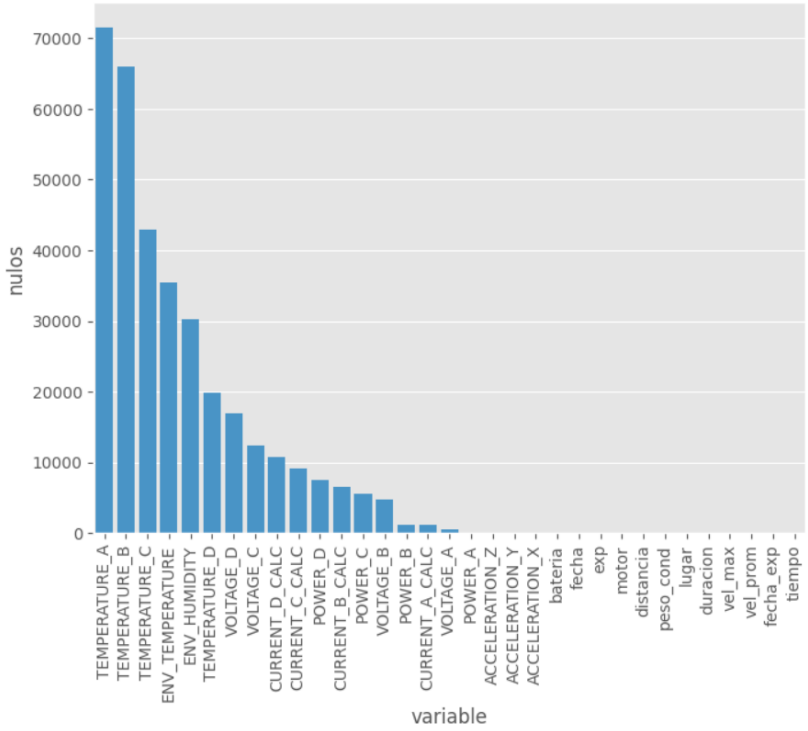
Figura 1
Histograma de los registros por ciclo



Posteriormente, se observa la cantidad de registros nulos por variable (**Figura *2***), encontrando casos con una alta concentración de faltantes, donde destacan TEMPERTURE_A, TEMPERATURE_B y TEMPERATURE_C. Más adelante se muestra las decisiones tomadas sobre imputación de nulos y eliminación de registros efectuadas con el fin de evitar inconvenientes relacionados con los registros nulos.

Figura 2

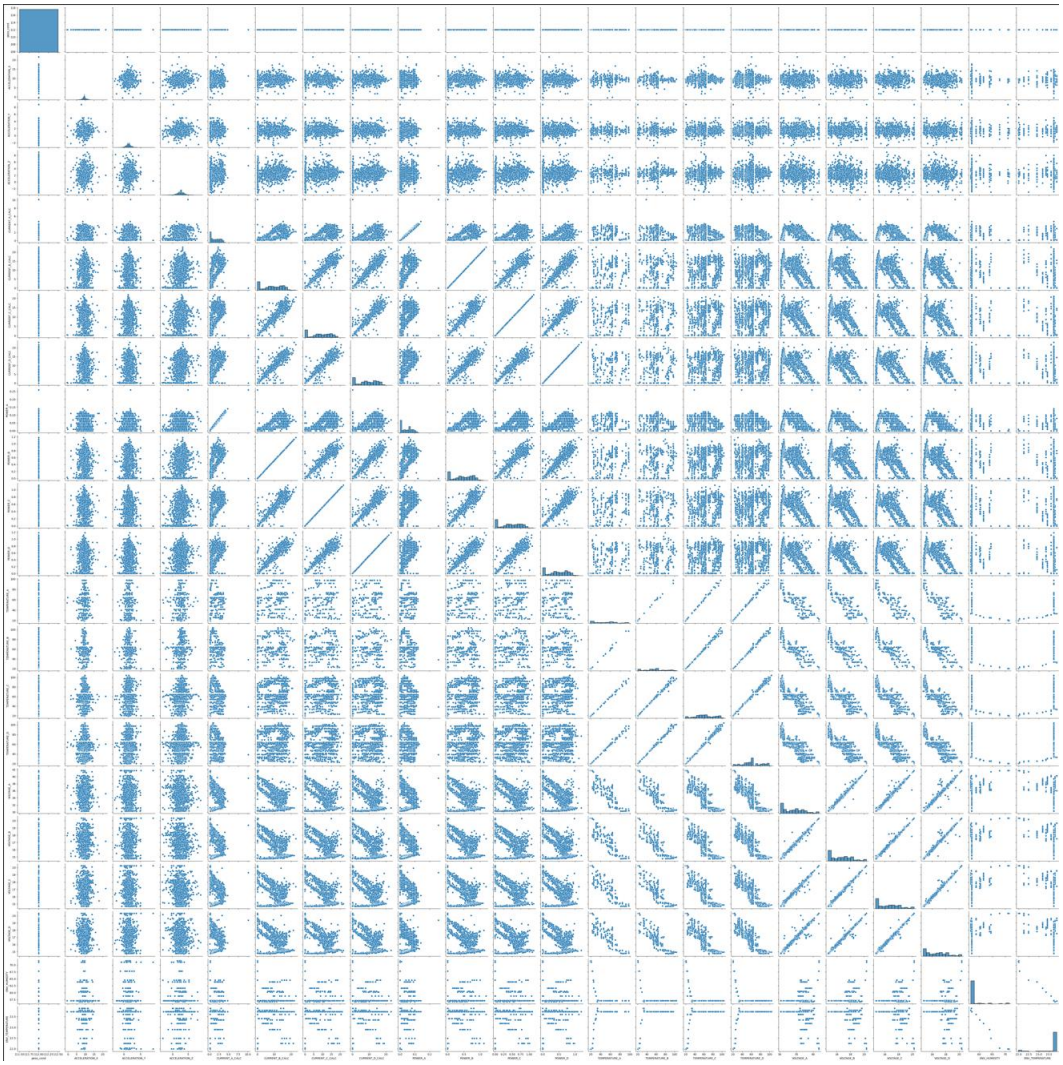
Diagrama de barras de los registros nulos por cada variable



Se exploró la distribución de las distintas variables del conjunto de datos generando pairplots para observar también posibles correlaciones entre las características mencionadas. A modo de ejemplo, se ilustra el resultado del ciclo 20221002_E_1_B01 (**Figura *3***) con las diferentes variables. Se puede observar relaciones lineales bastante marcadas entre ciertas variables. Estas relaciones se dan generalmente en conjuntos de variables como los de temperatura y voltaje. Estos gráficos también permiten identificar ciclos que hayan presentado falta de información en ciertas variables o comportamientos erráticos en estas.

Figura 3

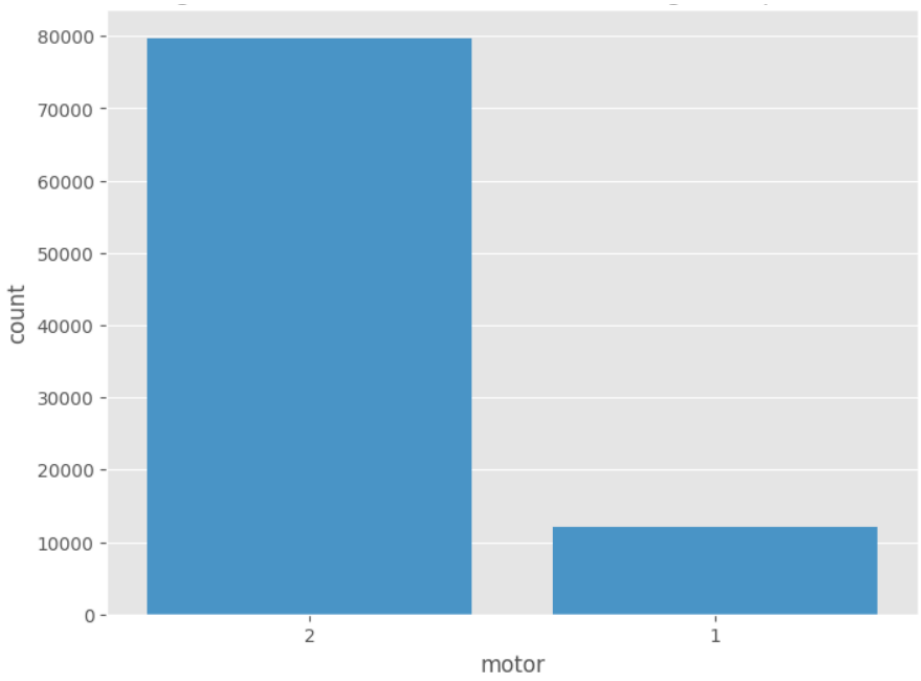
Ejemplo de pairplot implementado en el ciclo 20221002_E_1_B01



En cuanto a las variables categóricas, se exploraron los registros, posibles sesgos asociados a la disponibilidad de información de categorías específicas, comportamientos asociados, entre otros. Empezando por el caso del motor, se evidenció un alto desbalance entre los dos motores usados (**Figura *4***). Esto podría resultar muy problemático en un modelo de clasificación que use esta variable como respuesta. Sin embargo, en este caso, se evalúa el impacto con los investigadores del proyecto y se tomó la sugerencia de omitir dicha variable, puesto que se asume un comportamiento similar y se da prioridad a la información de la batería.

Figura 4

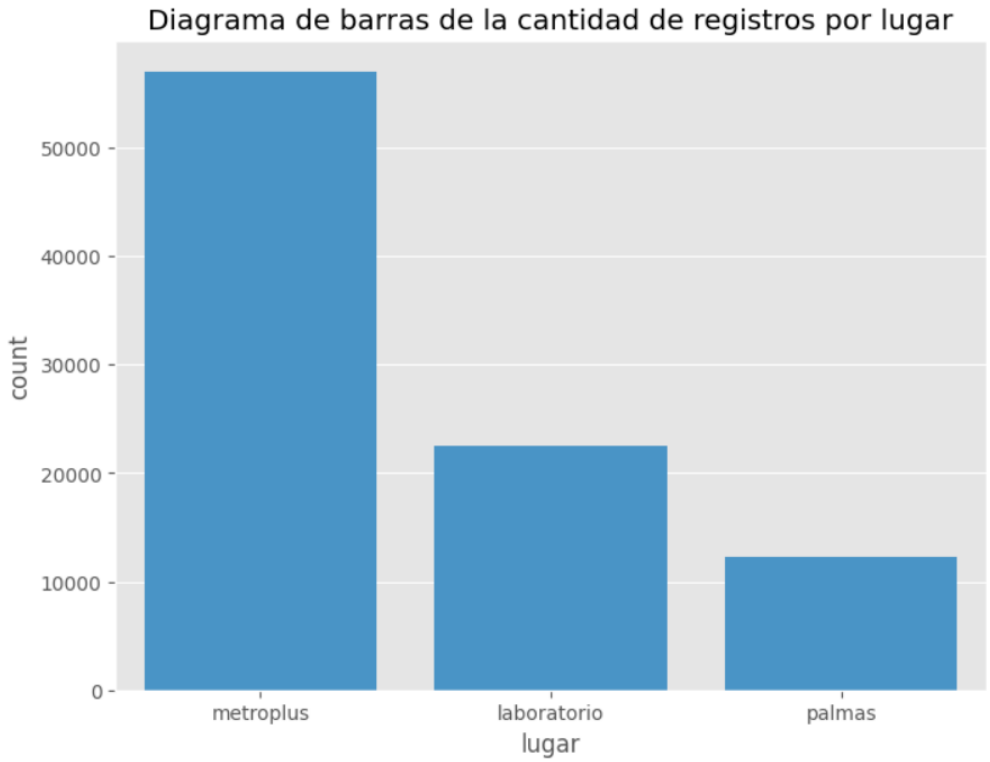
Diagrama de barras de los registros por motor



En cuanto al lugar del experimento, se evidenció que principalmente se efectuaron las pruebas en el entorno de vías de metroplus, seguido por el laboratorio y la vía Las Palmas en la ciudad de Medellín, Colombia (**Figura *5***). Las condiciones experimentales dificultaron la toma de información. Por ejemplo, la información de laboratorio no incluyó distancia ni velocidad promedio de la prueba. Las pruebas con mayor certeza de calidad de acuerdo con la información suministrada por los investigadores, se tomaron en las vías de metroplus, lo que hace decantarse por esta opción como prioridad al momento de seleccionar información.

Figura 5

Diagrama de barras de los registros por lugar de prueba



Algunas variables contienen información por ciclo como vel_max y vel_prom, es decir, se componen por un valor único para el ciclo completo. Bajo estas circunstancias, podría implicar que dichas variables no presentan variabilidad relevante que aporte información a los modelos. Sin embargo, son variables que describen de manera general el ciclo al que pertenece cada observación, lo que contrariamente a lo esperado, podría aportar variabilidad importante y convertirse en factores decisivos. Estas variables presentaron distribuciones no normales como puede apreciarse en la **Figura *6*** y **Figura *7***.

Figura 6

Histograma de la velocidad máxima

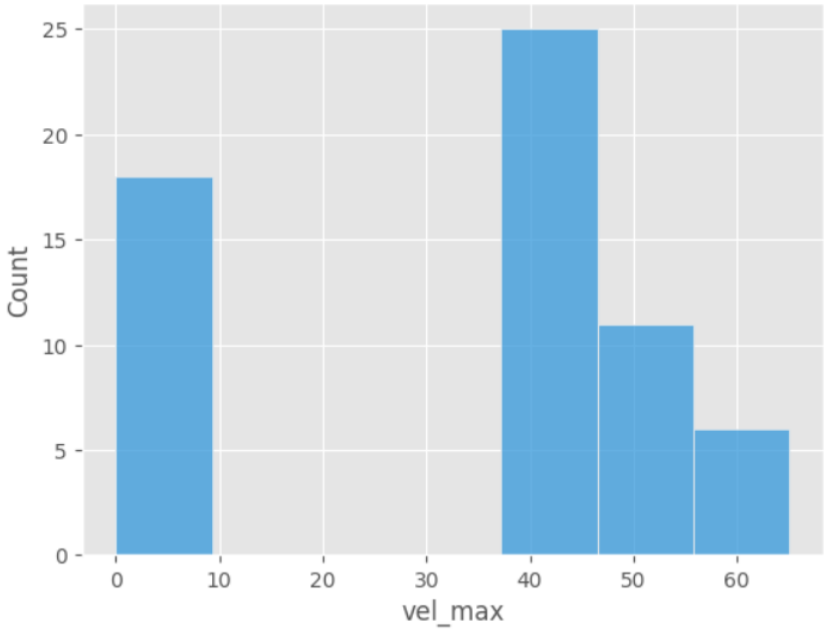
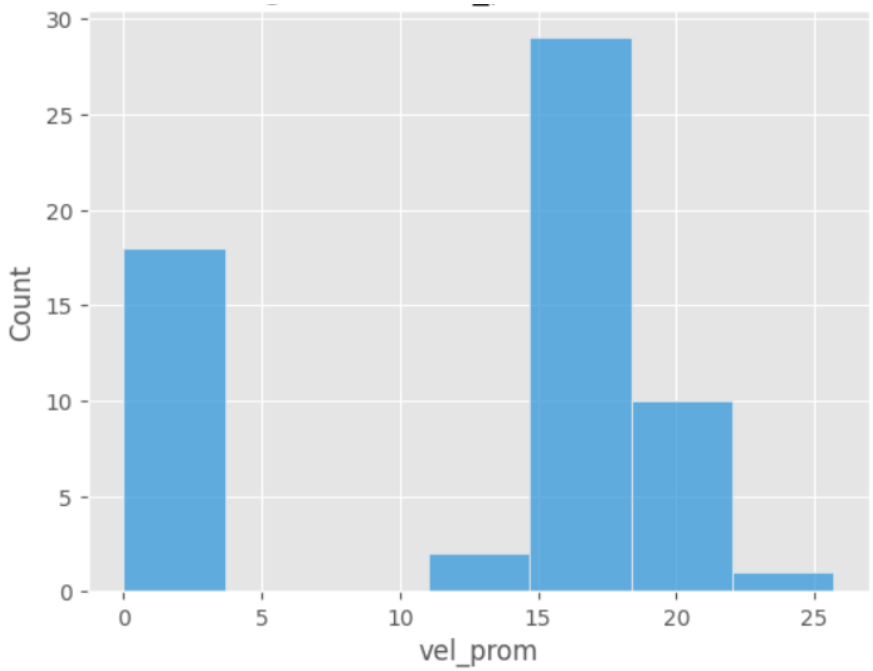


Figura 7

Histograma de la velocidad promedio

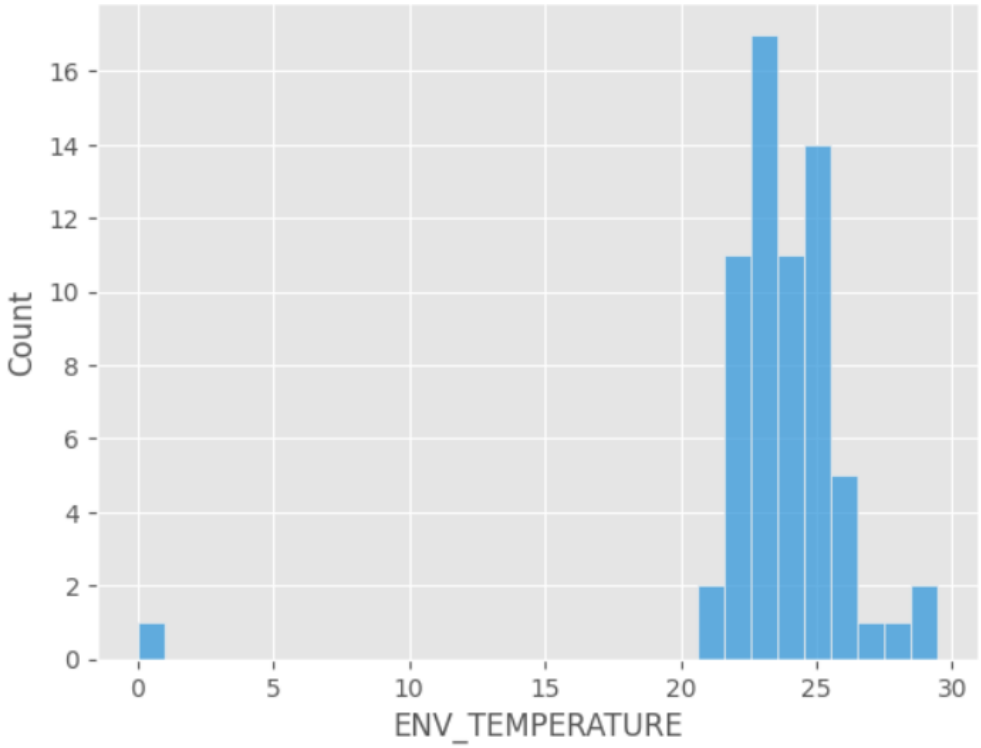


En cuanto a variables extrínsecas al sistema, se pudo analizar la temperatura y la humedad ambientales.

La temperatura ambiental muestra algunos registros en 0, mientras que la mayoría de los registros restantes, parece tener una distribución más acorde con un comportamiento centrado en una media (**Figura *8***). La distribución de la temperatura ambiental no es el enfoque de este ejercicio por lo que no se profundiza en su distribución esperada.

Figura 8

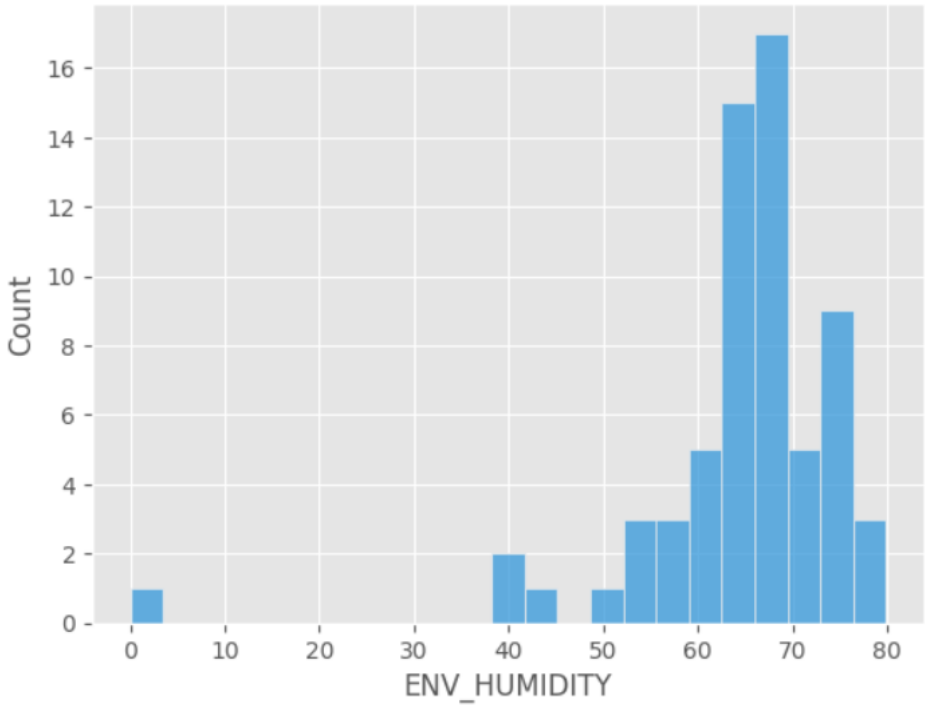
Histograma de la temperatura ambiente



La humedad ambiental, que puede verse en la **Figura *9***, presenta, al igual que en el caso anterior, una distribución que podría estar acorde con lo esperado de la medida en cuestión. Esta variable parece centrarse sobre una media cercana a 50, presentando algunos casos atípicos con valores significativamente bajos. Este histograma de la humedad ambiental en los ciclos, permite inferir que probablemente se tenga un comportamiento centrado en una media similar a una distribución normal con ciertas desviaciones.

Figura 9

Histograma de la humedad ambiental



Cabe resaltar que no todas las variables del ejercicio serán tenidas en cuenta, por diversas razones que se exploran en la preparación de los datos, puesto que los expertos consideraron que no aportan información relevante, se presenta una nulidad significativa o se tiene prioridad por una variable que aborda el problema de forma más clara.

7. Preparación de los datos

7.1 Limpieza de los datos

En primer lugar, se eliminaron algunos ciclos por decisión del equipo involucrado, teniendo en cuenta peculiaridades relacionadas con dificultades en la toma de muestras, incluyendo problemas ambientales, variaciones en las rutas establecidas, ausencia de información de temperatura, pocos registros, entre otras. Con esto presente, se eliminaron los ciclos: '20220916-E01', '20220918_E_2_B01', '20220920_E_1_B01', '20221024_E_2_B01', '20221025_E_1_B01', '20221017_E_2B01', '20221011_E_2_B01', '20221011_E_1_B01', '20221010_E_2_B01' y '20221029-E01'. En total, en este paso se eliminaron 9300 registros lo que representa

aproximadamente un 10% de la información disponible inicialmente. Cabe resaltar que la decisión de eliminación corresponde a consideraciones estratégicas por parte de los analistas involucrados.

Posteriormente, se procede con la limpieza de las variables, empezando por los voltajes. Para estos elementos, se exploraron los outliers desde una perspectiva clásica, considerando los límites inferior y superior de un diagrama de cajas y bigotes, es decir, los elementos por encima de $P_{75} + 1{,}5 * IQR$. Este valor se calculó para cada voltaje y se determinó que la variable VOLTAGE_A sería la guía de selección. Se consideraron también otras variables como el tiempo, de modo que la variable voltaje no presente un comportamiento anómalo.

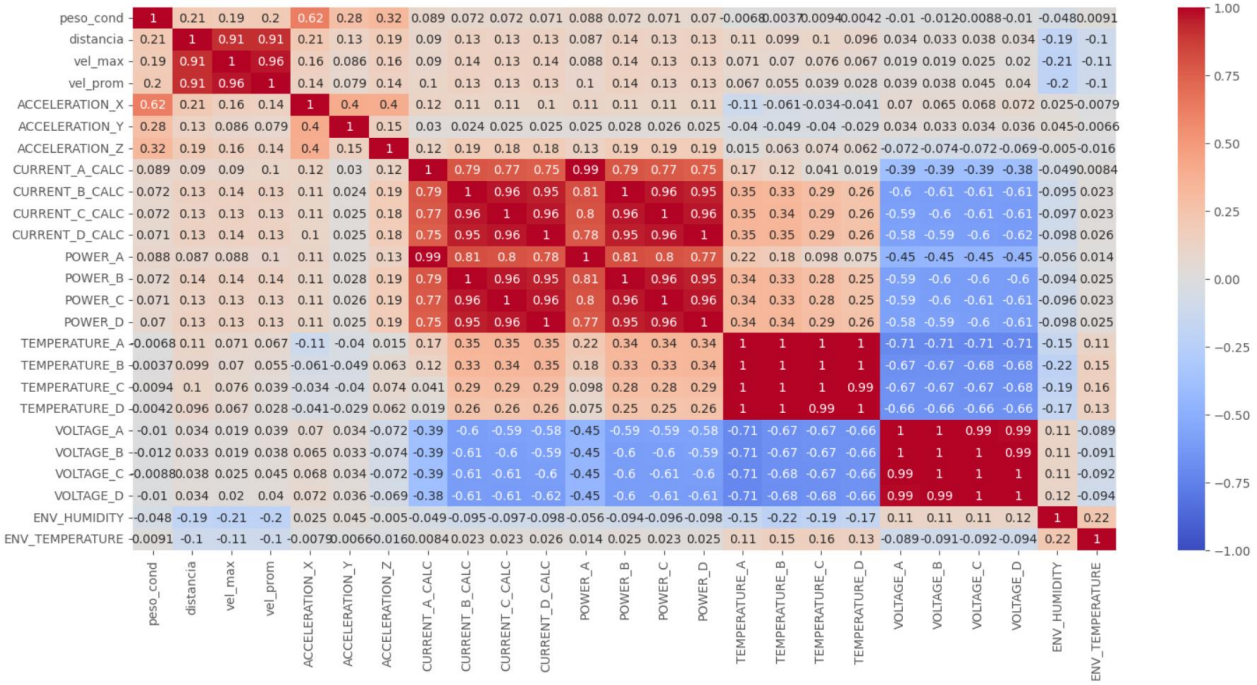
Luego, se añadió un contador para los registros considerando la eliminación de ciertos registros con información errónea o anómala. Continuando con el proceso, se convirtieron a nulas algunas temperaturas con valores inferiores a 15°C y superiores a 120°C en los puntos del sistema.

Con esta transformación, el conjunto de datos pasó a contener 78.886 registros con 34 características. Con este conjunto de datos, se analizaron las correlaciones existentes con el fin de eliminar variables que puedan presentar redundancia a los modelos.

Se encontraron correlaciones muy marcadas (**Figura *10***), principalmente en las medidas de elementos similares entre sí, como el caso de los voltajes, las temperaturas y la corriente. Con el fin de estipular las variables a usar, se implementó un algoritmo de minimización del VIF que arrojó la conveniencia de las variables: 'TEMPERATURE', 'distancia', 'vel', 'POWER', 'ENV', 'CURRENT', 'VOLTAGE', 'ACCELERATION'. Sin embargo, esta minimización del problema no resulta conveniente para los analistas, puesto que se eliminaría del modelo múltiples variables que pueden aportar interpretabilidad, ser relevantes para los fabricantes e investigadores e incluso, mejorar el desempeño de los modelos implementados.

Figura 10

Correlograma de las variables numéricas



Se realizó un análisis ANOVA y kruskal wallis para identificar similitudes en las variables, que permitan su eliminación. Sin embargo, estas pruebas reportaron diferencias dentro de cada grupo respectivo. Se optó por conservar las variables en su totalidad y seleccionar desde la perspectiva de los analistas las que se consideraran relevante e incluso evaluar las importancias de las variables a través de los modelos implementados.

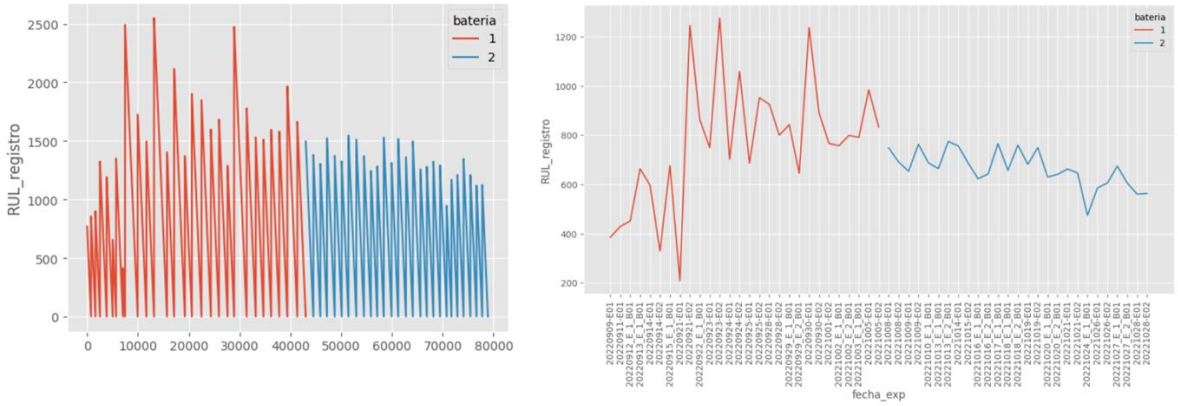
Posteriormente se realizó una imputación de registros en las variables 'ACCELERATION_X', 'ACCELERATION_Y', 'ACCELERATION_Z', 'CURRENT_A_CALC', 'CURRENT_B_CALC', 'CURRENT_C_CALC', 'CURRENT_D_CALC', 'POWER_A', 'POWER_B', 'POWER_C', 'POWER_D', 'TEMPERATURE_A', 'TEMPERATURE_B', 'TEMPERATURE_C', 'TEMPERATURE_D', 'VOLTAGE_A', 'VOLTAGE_B', 'VOLTAGE_C', 'VOLTAGE_D', 'ENV_HUMIDITY', y 'ENV_TEMPERATURE'. Esta imputación se realizó usando el algoritmo interpolate de pandas, agrupando por cada ciclo, logrando, además, mantener un comportamiento similar al previo en las variables imputadas. Por último, se añadió una imputación por vecinos cercanos con el paquete scikit-learn para las variables de ‘VEL_PROM’, ‘distancia’ y ‘VEL_MAX’.

7.2 Variable objetivo

Se partió de una construcción del RUL que permitiera analizar todos los registros. Para ello, se consideró la falla como la descarga de la batería y el RUL como el tiempo entre fallas. Adicionalmente, es importante considerar mantener el recuento de ciclo por cada batería para analizar el deterioro de estas y de forma más clara, la vida útil remanente. El RUL por registro presentó el comportamiento evidenciado en la **Figura 11** (costado izquierdo). Este RUL, promediado por ciclo se puede apreciar al costado derecho de la figura.

Figura 11

Diagrama de líneas en el tiempo del RUL, segmentado por batería (izquierda) y RUL promediado por ciclo (derecha)



7.3 Datasets

Por decisión del equipo de analistas se realizó un filtrado final de los registros considerando los siguientes elementos: Se tomarían únicamente los registros tomados en las vías de metroplus, aunque conservando los números de ciclo previamente calculados.

Con este conjunto final de datos se consideraron 46.781 registros con 25 variables: 'distancia', 'vel_max', 'vel_prom', 'ACCELERATION_X', 'ACCELERATION_Y', 'ACCELERATION_Z', 'CURRENT_A_CALC', 'CURRENT_B_CALC', 'CURRENT_C_CALC', ‘CURRENT_D_CALC', 'POWER_A', 'POWER_B', 'POWER_C', 'POWER_D', 'TEMPERATURE_A', 'TEMPERATURE_B', 'TEMPERATURE_C', 'TEMPERATURE_D', 'VOLTAGE_A', 'VOLTAGE_B', 'VOLTAGE_C', 'VOLTAGE_D', 'ENV_HUMIDITY', 'ENV_TEMPERATURE' y 'ciclo'.

Mediante el paquete scikit-learn se implementaron algunos modelos estándar que permitieron analizar el peso de las variables involucradas. Esto, apoyado de la decisión de un equipo de expertos conformado por investigadores relacionados con el proyecto, permitió confirmar las variables a tener en cuenta. Una primera etapa exploró la aplicación de las variables incluyendo el ciclo (***Tabla 2***), y una segunda etapa sin incluirlo (***Tabla 3***). Se utilizaron los modelos Random Forest Regressor (RFR), GBR y ABR.

Tabla 2

Pesos de variables incluyendo la variable ciclo para RFR, GBR y ABR

Variables	Pesos con RFR*	Pesos con GBR**	Pesos con ABR***
VOLTAGE_A	0,1866373	0,2311048	0,1699221
VOLTAGE_B	0,1622612	0,1739654	0,0969401
TEMPERATURE_A	0,1520546	0,0950438	0,0935556
VOLTAGE_C	0,1160372	0,0844928	0,0472998
distancia	0,0784685	0,0786913	0,0840468
ciclo	0,0598862	0,0505479	0,0558978
vel_prom	0,0486672	0,0497266	0,0553314
VOLTAGE_D	0,0477902	0,048823	0,0863886
TEMPERATURE_B	0,0313106	0,0387598	0,0369999
TEMPERATURE_C	0,0258073	0,0348292	0,0821755
ENV_HUMIDITY	0,0170685	0,0200924	0,0314283
TEMPERATURE_D	0,0127436	0,0173637	0,0235275
ENV_TEMPERATURE	0,0112446	0,0149576	0,0170785
CURRENT_A_CALC	0,0090365	0,011176	0,0349396
POWER_B	0,0068295	0,0094194	0,0158915
vel_max	0,006633	0,0091276	0,0210972
CURRENT_C_CALC	0,0058465	0,0087803	0,0099559
ACCELERATION_X	0,0057947	0,006687	0,0074015
CURRENT_B_CALC	0,0054169	0,0051713	0,009581
CURRENT_D_CALC	0,0046936	0,004131	0,0094891
POWER_C	0,0028292	0,0032935	0,0048116
POWER_D	0,0016142	0,0025615	0,0062407
ACCELERATION_Y	0,0004811	0,0005011	0
POWER_A	0,0004274	0,0003809	0
ACCELERATION_Z	0,0004204	0,0003721	1,257E-10

*(n_estimators=500). RMSE = 32,64, R2 = 0,9945

**(learning_rate=0.2, max_depth=7, n_estimators=220), RMSE = 38,73, R2 = 0,9922

***(learning_rate=0.9, loss='square', n_estimators=180), RMSE = 12,89, R2 = 0,8570

Tabla 3

Pesos de variables sin la variable ciclo para RFR, GBR y ABR

Variables	Pesos con RFR*	Pesos con GBR**	Pesos con ABR***
VOLTAGE_A	0,1859807	0,2675168	0,0805916
VOLTAGE_B	0,1657291	0,1691079	0,0301078
TEMPERATURE_A	0,1513638	0,093412	0,0843965
VOLTAGE_C	0,1224261	0,0232429	0,0424371
distancia	0,0942423	0,0989597	0,1330285
vel_prom	0,0646405	0,0551757	0,2168001
VOLTAGE_D	0,034258	0,0676148	0,0525515
TEMPERATURE_B	0,028648	0,0438031	0,0849035
ENV_TEMPERATURE	0,0284035	0,0203845	0,0402305
ENV_HUMIDITY	0,0281976	0,0214671	0,0195723
TEMPERATURE_C	0,0258231	0,0477128	0,0414914
vel_max	0,0173071	0,0198683	0,0494904
TEMPERATURE_D	0,0132537	0,0123832	0,0402015
CURRENT_A_CALC	0,0089415	0,0189469	0,0292556
ACCELERATION_X	0,0063185	0,0046395	0
POWER_B	0,0053877	0,0079515	0,0152331
CURRENT_B_CALC	0,0052115	0,0074635	0,0155959
CURRENT_C_CALC	0,0046872	0,0065927	0,0088794
CURRENT_D_CALC	0,0045396	0,0050274	0,0030908
POWER_C	0,0020028	0,0040573	0,0001502
POWER_D	0,0012795	0,0031123	0,0102448
ACCELERATION_Y	0,0004902	0,0004705	0
ACCELERATION_Z	0,0004478	0,0004283	7,106E-06
POWER_A	0,0004204	0,0006614	0,0017405

*(n_estimators=500). RMSE = 32,24, R2 = 0,9946

**(learning_rate=0.2, max_depth=7, n_estimators=220), RMSE = 38,76, R2 = 0,9922

***(learning_rate=0.9, loss='square', n_estimators=180), RMSE = 184,50, R2 = 0,8238

Se seleccionaron las variables: 'VOLTAGE_B', 'VOLTAGE_A', 'TEMPERATURE_A', 'TEMPERATURE_B', 'TEMPERATURE_C', 'distancia', 'vel_prom', 'POWER_B', 'CURRENT_B_CALC', 'ENV_HUMIDITY', 'ENV_TEMPERATURE', 'CURRENT_A_CALC', 'POWER_A'. La variable de batería se usó exclusivamente para mantener equilibrado el conjunto de datos de entrenamiento y prueba (mediante la librería imblearn y la función RandomUnderSampler), pero se eliminó para la aplicación de los modelos.

Para el conjunto de prueba, se seleccionó de forma aleatoria una representación homogénea de las 2 baterías, de la cual se extrajo el 70% de los registros mediante la función train_test_split del paquete scikit-learn. El 30% restante se dividió nuevamente en 70% de prueba 1 y 30% restante de prueba 2. Es decir, del 30% de la primera extracción se dejó aproximadamente el 20% para la primera prueba y el 10% restante para la segunda.

8. Construcción de modelos de regresión

Se consideraron modelos de constante aplicación práctica que permitieran cierta interpretabilidad con respecto a los resultados y la importancia de las variables involucradas. Cada modelo pasó por una etapa exploratoria y una etapa de Grid Search CV.

8.1. SVR

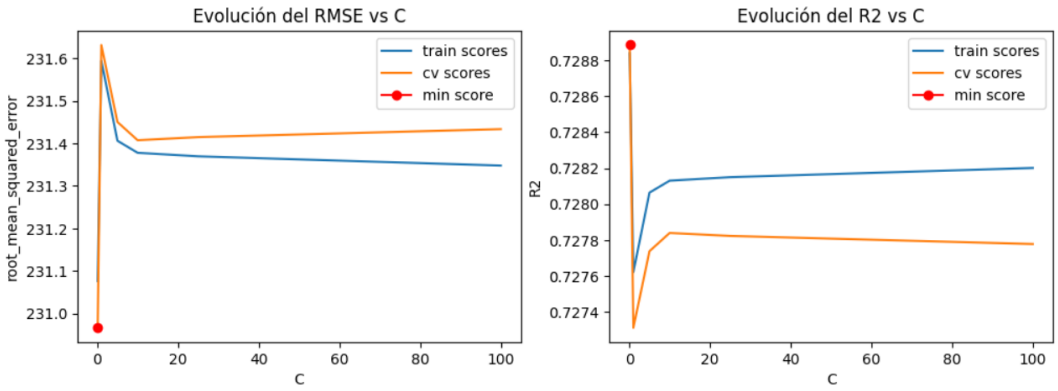
La máquina de vectores de soporte, enfocada en regresión se ajustó con respecto a los hiperparámetros C, kernel, gamma y degree.

Exploración de hiperparámetros

En primera instancia, se fijó el kernel como lineal y se exploró diferentes valores para C, analizando el desempeño tanto en el RMSE como en R2. El tiempo de procesamiento es muy elevado, implicando más de 40 minutos en la elaboración de la **Figura *12*** y el resultado obtenido no es significativo en cuanto a interpretación, puesto que se obtiene el mejor desempeño en la primera iteración del modelo, por lo que se opta por pasar directamente a la etapa de Grid search.

Figura 12

Exploración de valores de C para SVR con kernel lineal



Fuente: Elaboración propia.

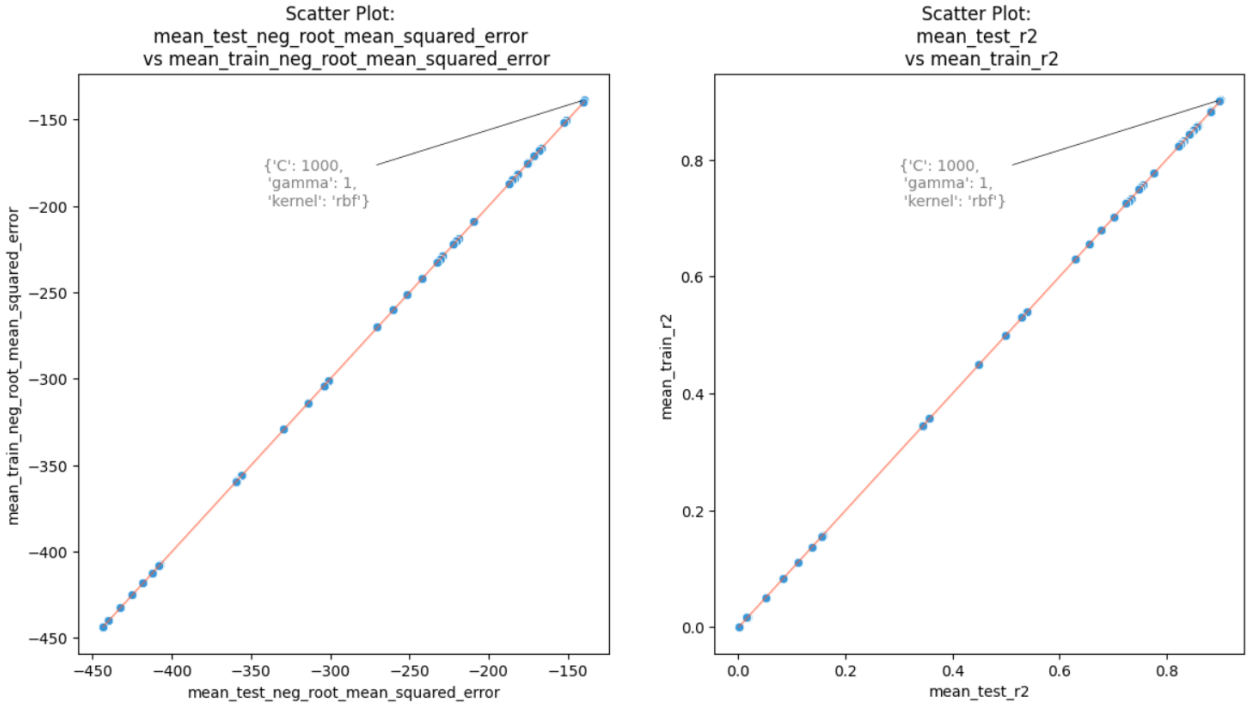
Grid search CV

La búsqueda de rejilla, o grid search se realizó con la combinación de múltiples hiperparámetros. El costo computacional del SVR suele ser elevado por lo que se buscó alternativas de aplicación y se optó por el paquete cuML de Nvidia (Raschka et al., 2020) , que provee la interfaz de scikit-learn pero se ejecuta a través de GPU. Se utilizó un computador con una tarjeta gráfica Nvidia RTX 4060 mobile.

Si bien el desempeño más elevado se presenta con C=1000, este hiperparámetro tiende a sesgar el modelo puesto que regulariza el rango de tolerancia, afectando la flexibilidad del modelo para ajustarse a variaciones naturales en los datos. Por lo anterior se optó por una medida más recogida y con un desempeño cercano: C=10, degree=2, gamma=0,01, kernel=’poly’ **Figura *13***.

Figura 13

Resultados de RMSE y R2 para distintos modelos SVR



Fuente: Elaboración propia.

8.2 GBR

En el caso de Gradient Boost Regressor, se exploraron los hiperparámetros de learning_rate, n_estimators y max_depth. Posteriormente se ejecutó el grid search cv para encontrar el modelo de mejor desempeño.

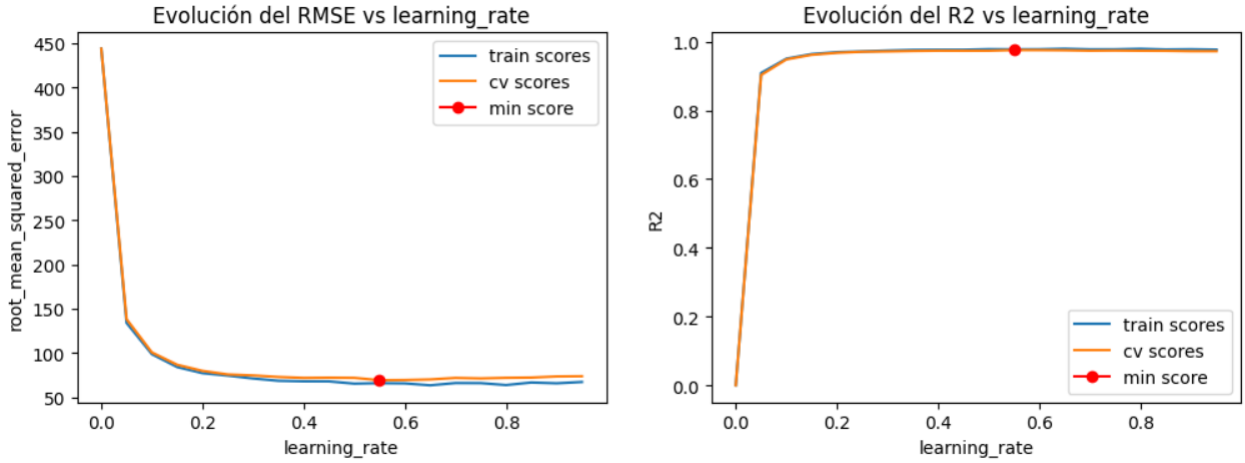
Exploración de hiperparámetros

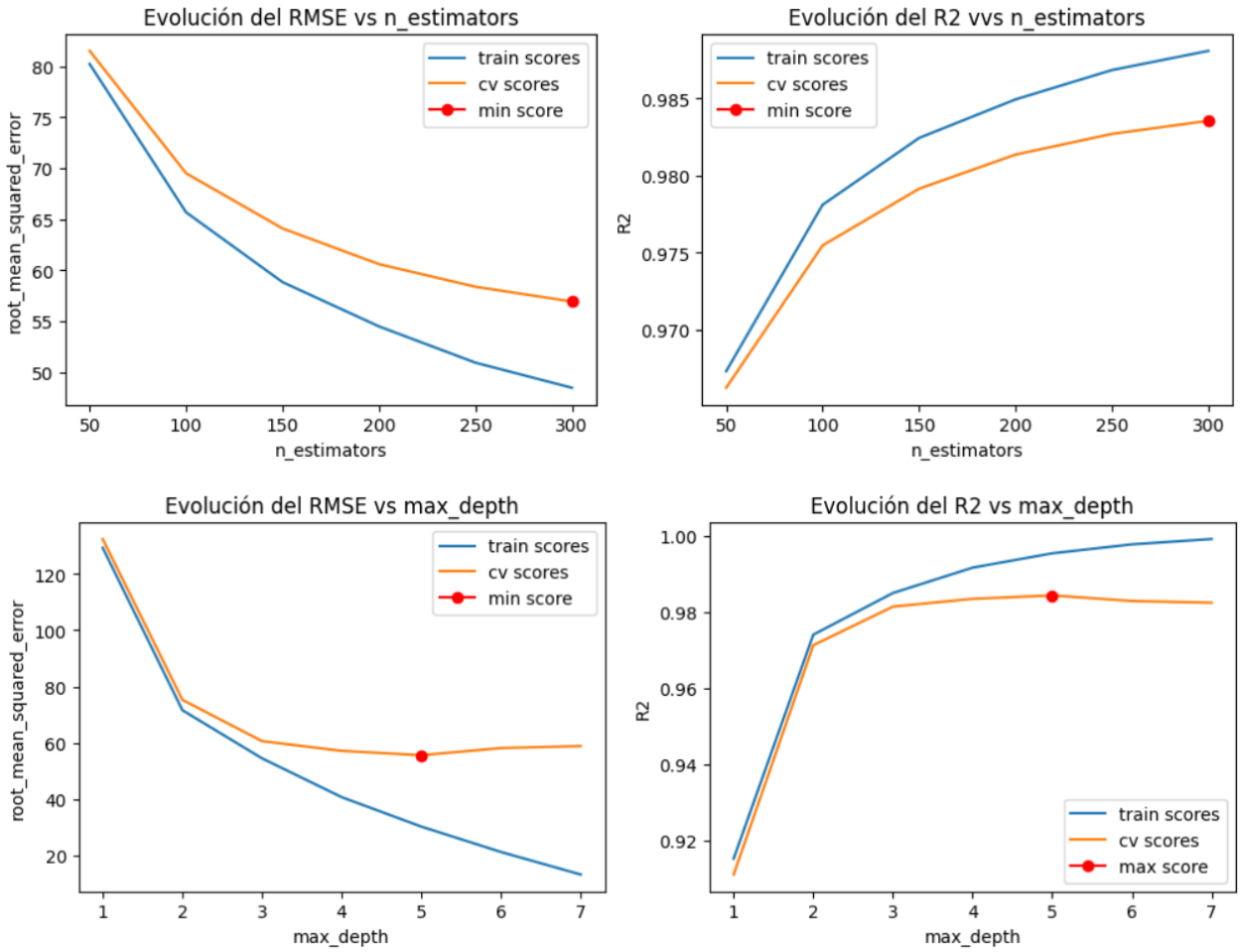
Se exploraron los hiperparámetros fijando algunos elementos y variando otros.

Como se observa en la **Figura *14***, se evidenció cierto estancamiento del desempeño para learning_rate y max_depth, puesto que las métricas no presentan mejoras evidentes posteriores a 0,5. Por otra parte, se evidencia una aparente mejora continuada en las métricas al aumentar la cantidad de estimadores (n_estimators). Finalmente, se evidenció cierto nivel de overfit para valores muy altos de profundidad (max_depth), puesto que las métricas mejoran para el conjunto train pero empeoran para el test.

Figura 14

Evaluación de tasa de aprendizaje, estimadores y profundidad en GBR





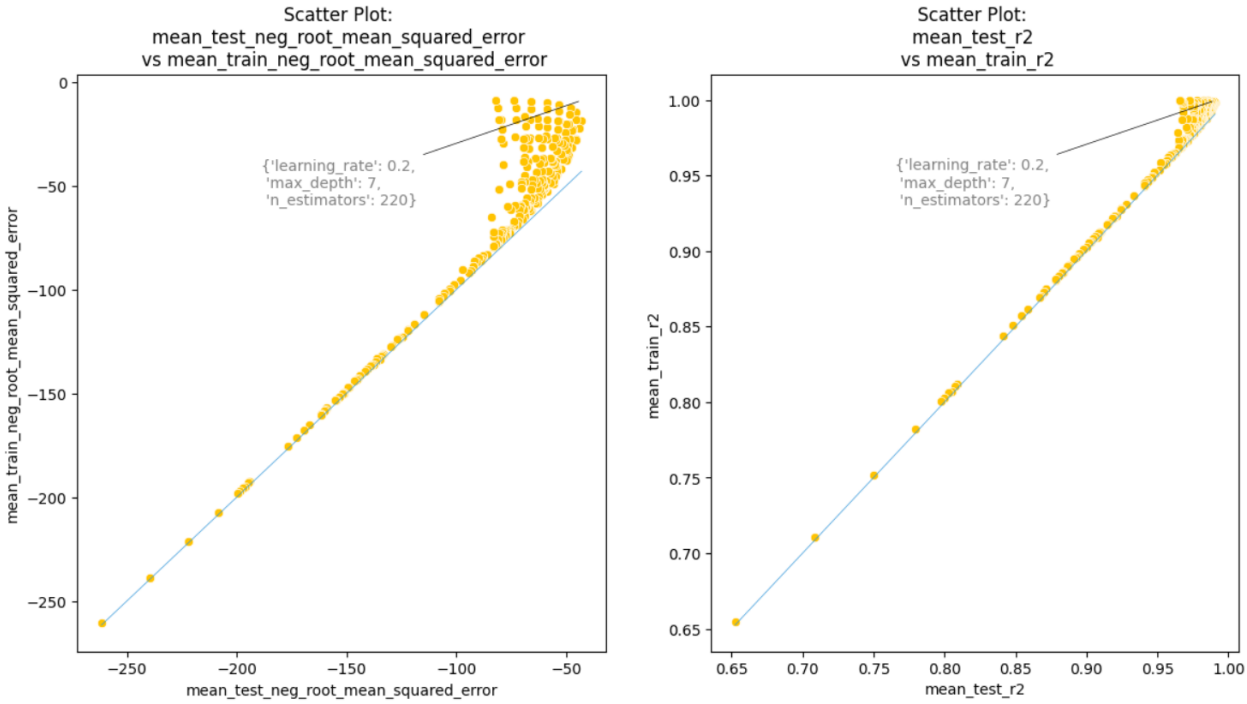
Fuente: Elaboración propia.

Grid search CV

En la búsqueda de rejilla se identificó un modelo con learning_rate=0,2, max_depth=7 y n_estimators=220 como el de mejor desempeño dado un RMSE de 43 y R2 de 0,99. Este modelo se eligió para las pruebas posteriores. El modelo se ubica en la esquina superior derecha de los gráficos de la **Figura *15***.

Figura 15

Resultados de RMSE y R2 para distintos modelos GBR



Fuente: Elaboración propia.

8.3 ABR

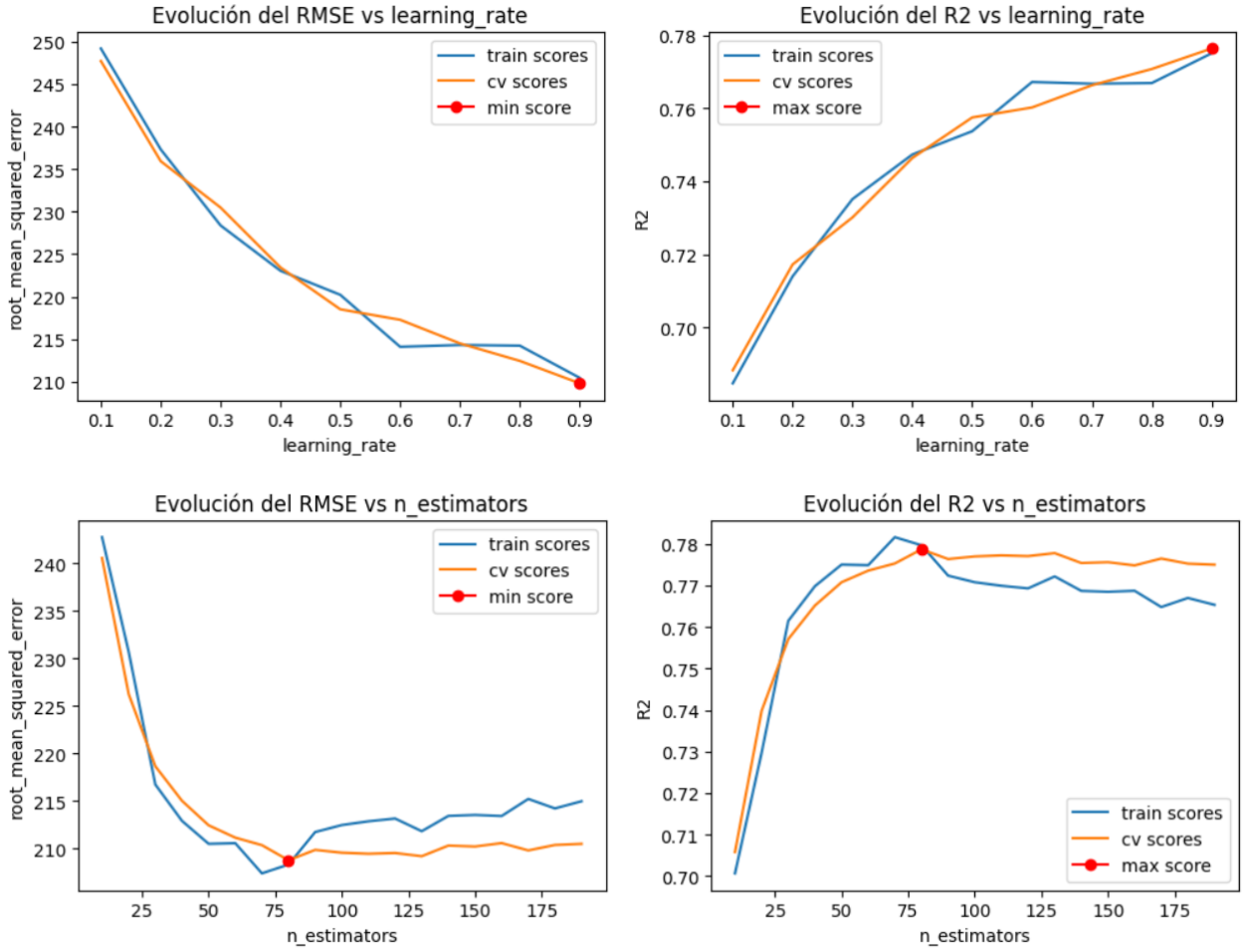
El modelo AdaBoost Regressor se exploró de forma similar al GBR. En este caso, se evaluaron los parámetros de learning_rate y n_estimators.

Exploración de hiperparámetros

En cuanto al comportamiento gráfico, en la **Figura *16*** se observa una aparente mejora constante en las métricas con la tasa de aprendizaje aumentando, pero un estancamiento de la cantidad de estimadores, donde se presenta un mejor desempeño en una cantidad cercana a 80 estimadores. En ese orden de ideas, podría sugerirse una tasa de aprendizaje elevada, manteniendo controlada la cantidad de estimadores. Estas posibilidades se optimizan en cuanto al desempeño en la etapa de grid search.

Figura 16

Evaluación de tasa de aprendizaje y estimadores para ABR



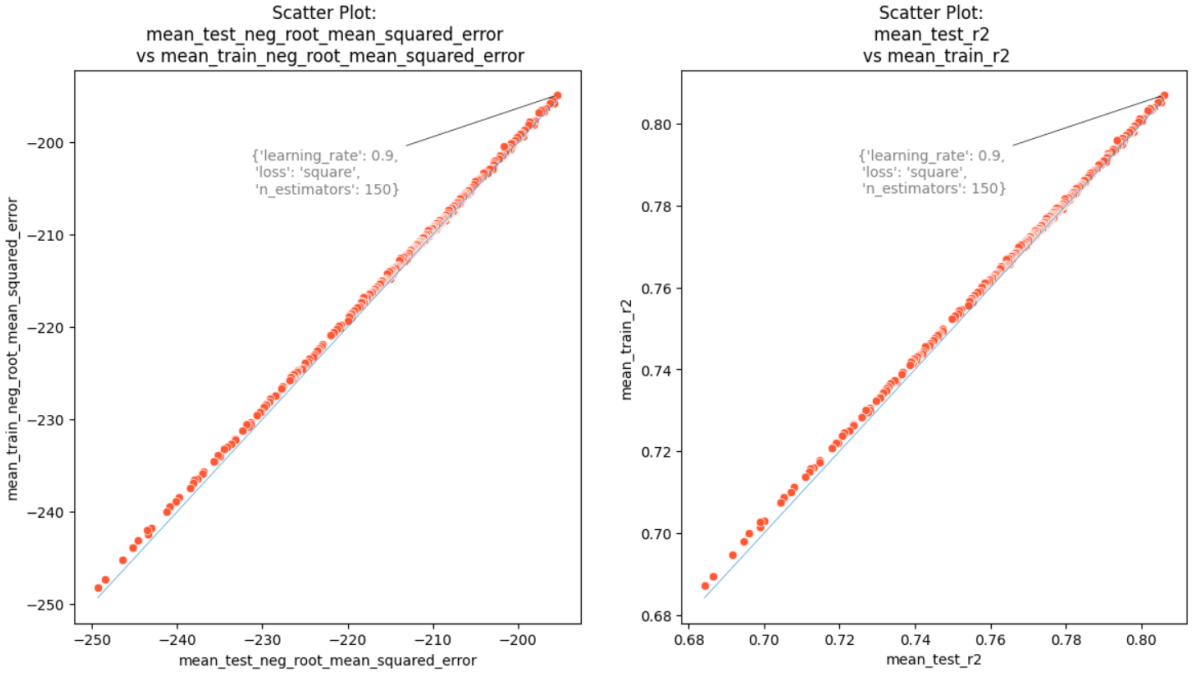
Fuente: Elaboración propia.

Grid search CV

En cuanto a la búsqueda de rejilla, se eligió un modelo con learning_rate=0,9, loss=’square’ y n_estimators=150, dado su alto desempeño tanto en RMSE (195,38) como en R2 (0,81) (**Tabla *4***). Este modelo se ubica en la esquina superior derecha de los gráficos en la **Figura *17***.

Figura 17

Resultados de RMSE y R2 para distintos modelos ABR



Fuente: Elaboración propia.

9. Resultados y discusión: evaluación y selección de modelos

Los modelos elegidos por su desempeño y/o características en el grid search dentro de cada tipo de modelo implementado (GBR, ABR y SVR) se muestran en la **Tabla *4*¡Error! No se encuentra el origen de la referencia.**. El modelo GBR muestra un desempeño claramente superior. Los casos de SVR de mejor desempeño presentaban cierto overfit por lo que se eligió este modelo que presenta un desempeño muy inferior al GBR y ABR.

Tabla 4

Modelos, parámetros y métricas de desempeño en Grid search de los modelos elegidos

Modelo	Parámetros	RMSE promedio test	RMSE desviación test	R2 promedio test	R2 desviación test	RMSE promedio train	RMSE desviación train	R2 promedio train	R2 desviación train
GBR	{'learning_rate': 0.2, 'max_depth': 7, 'n_estimators': 220}	-43,001	1,091	0,991	0,001	-18,796	0,282	0,998	0
ABR	{'learning_rate': 0.9, 'loss': 'square', 'n_estimators': 150}	-195,384	2,500	0,806	0,005	-194,901	2,462	0,807	0,005
SVR	{'C': 10, 'degree': 2, 'gamma': 0.01, 'kernel': 'poly'}	-301,083	2,649	0,54	0,006	-301,056	0,566	0,54	0,002

9.1. Cross-Validation

Con los modelos elegidos en la etapa 8 (**Tabla *4***), se realizó una validación cruzada, usando 5 pliegues, en los que se integró el conjunto de prueba inicial y el primer conjunto de testeo. Se evaluaron las métricas de RMSE, MSE, MAE, MAPE y R2.

*En todos los casos predomina el desempeño del modelo GBR (**Figura* **18***), presentando un RMSE promedio para test de 40,3. Adicionalmente, R2 para este modelo es de 0,99 lo que le permite explicar una gran proporción de la variabilidad del RUL. Esto puede verse con precisión en la*

Tabla *5*. La ***Figura 18*** también muestra diagramas de cajas que evidencian diferencias estadísticas entre los resultados de los modelos probados. En todos los casos, se evalúa los valores más altos de las métricas, teniendo en cuenta que, como se mencionó previamente, las métricas que convienen cuanto menores, se establecen negativas para facilitar la visualización.

Figura 18

Diagramas de cajas de las métricas de desempeño de prueba en Cross-Validation para los modelos elegidos

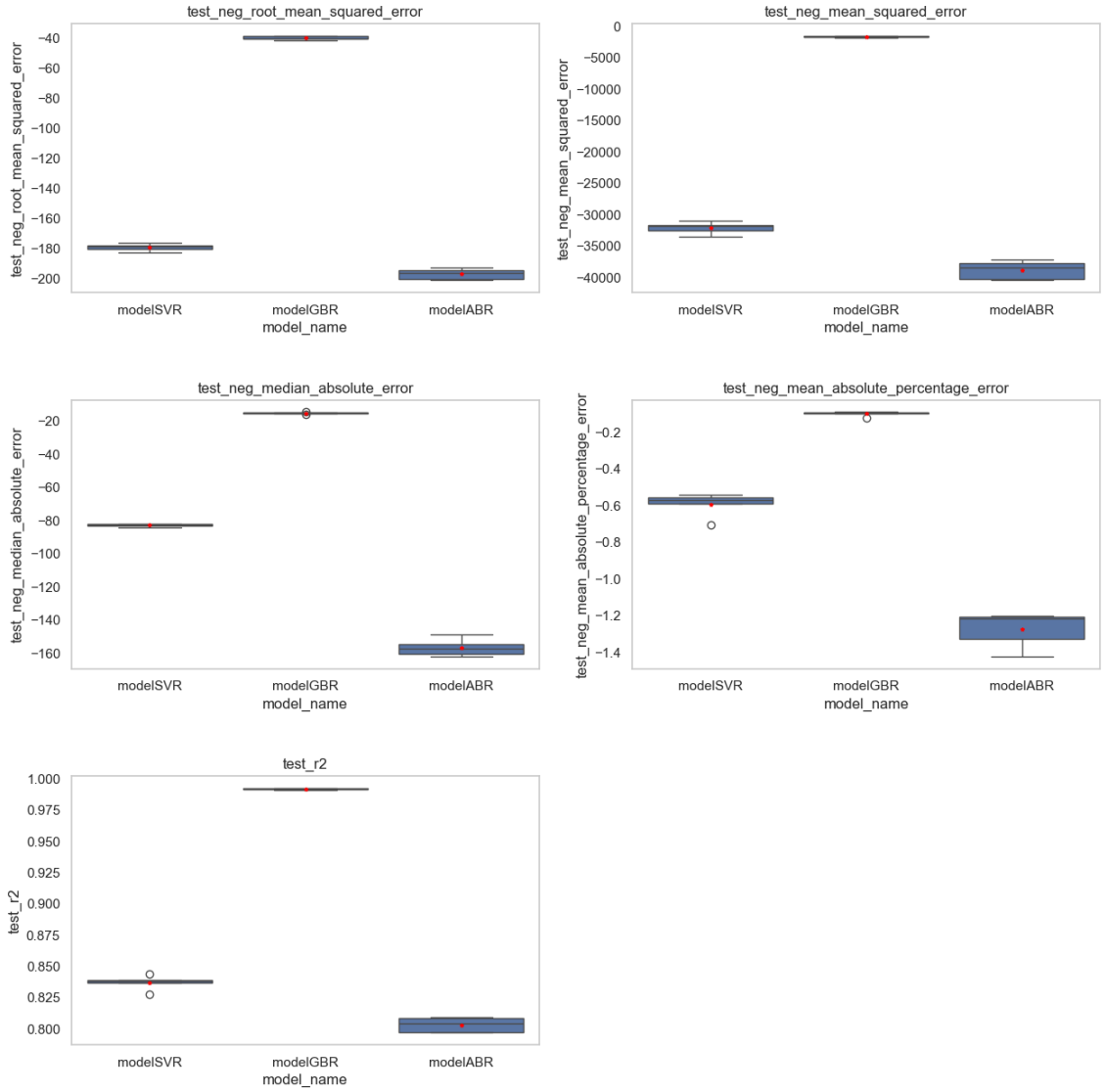


Tabla 5

Métricas de desempeño en validación cruzada para los modelos elegidos

	RMSE test		MSE test		MAE test		MAPE test		R2 test	
	Promedio	**Desviación**	**Promedio**	**Desviación**	**Promedio**	**Desviación**	**Promedio**	**Desviación**	**Promedio**	**Desviación**
Modelo										
GBR	-40,33	1,35	-1628,19	109,26	-15,32	0,46	-0,10	0,01	0,99	0,00
ABR	-197,01	3,77	-38824,21	1486,99	-156,61	5,21	-1,28	0,10	0,80	0,01
SVR	-179,26	2,57	-32140,53	925,44	-82,87	0,87	-0,59	0,07	0,84	0,01

9.2. Evaluación cualitativa

El RUL en este enfoque hacia confiabilidad, puede predecirse de una forma aceptable a través del modelo GBR propuesto. El error RMSE se presenta en las unidades del RUL. Dado que el RUL representa el tiempo hasta la descarga, existe una desviación muy pequeña del valor predicho en comparación con el valor real. Además, la variabilidad explicada es muy cercana a 1, lo que lo hace un modelo aparentemente robusto.

9.3. Consideraciones de producción

En esta última etapa, se realiza una última validación del modelo elegido con el conjunto final de prueba (test2), obteniendo un desempeño nuevamente superlativo (***Tabla 6***).

Tabla 6

Métricas de desempeño para el modelo final (GBR) con test2*

Medidas	Resultados
RMSE	6,15
MSE	37,80
MAE	23,57
MAPE	0,10
R2	0,99

*'learning_rate': 0.2, 'max_depth': 7, 'n_estimators': 220

El tiempo de entrenamiento (ajuste) de este modelo ronda los 40 segundos en un procesador Intel core i7 de 13ava generación. Mientras que la predicción toma 20 ms. Este modelo es eficiente. Sin embargo, cabe resaltar que las variables usadas en la construcción dependen de la instalación de sensores en zonas específicas de la batería, lo que podría ser costoso y difícil de ejecutar en la práctica, para mantener un seguimiento constante en los niveles de temperatura, voltaje, corriente, potencia, entre otros pero dado que permite determinar los tiempos de garantía, podría justificar la inversión en la instalación y recolección de información. Para precisar las variables cuya recolección se sugiere a partir de los pesos del modelo, conviene tener en cuenta: vel_max, distancia, vel_prom y ACCELERATION_Z principalmente. Los pesos obtenidos en el modelo final pueden apreciarse en la **Tabla *7***.

Tabla 7

Pesos de las variables en el modelo GBR final

Variables	Pesos
vel_max	0,286225
distancia	0,267567
vel_prom	0,111291
ACCELERATION_Z	0,102175
CURRENT_A_CALC	0,058928
ACCELERATION_X	0,055531
CURRENT_B_CALC	0,03409
CURRENT_D_CALC	0,019628
CURRENT_C_CALC	0,018161
ACCELERATION_Y	0,017718
POWER_B	0,016067
POWER_A	0,011669
POWER_C	0,000949

*'learning_rate': 0.2, 'max_depth': 7, 'n_estimators': 220

10. Conclusiones

El desempeño de los modelos de inteligencia artificial dependerá netamente del conjunto de datos. Ahí radica la importancia de la preparación y limpieza de la información disponible y la validación con expertos para establecer las variables adecuadas. Además, es fundamental y prioritario comprender la naturaleza del problema y la estructura de la información disponible, antes de proponer modelos o implementar estrategias estadísticas. En este caso, la decisión de los expertos se apoyó en múltiples herramientas cuantitativas que incluyen modelos estándar de inteligencia artificial con un desempeño que permita explicar un porcentaje superior al 70% de la variabilidad (R2). Para este caso en particular el uso de RFR, GBR y ABR como selectores de variables permitió una visual sobre la influencia de las variables y seleccionar las más adecuadas para la construcción de modelos más robustos. Las variables elegidas fueron: 'VOLTAGE_B', 'VOLTAGE_A', 'TEMPERATURE_A', 'TEMPERATURE_B', 'TEMPERATURE_C', 'distancia', 'vel_prom', 'POWER_B', 'CURRENT_B_CALC', 'ENV_HUMIDITY', 'ENV_TEMPERATURE', 'CURRENT_A_CALC', 'POWER_A'.

El ajuste de los hiperparámetros de los modelos es una etapa retadora que involucra el conocimiento de los efectos que los cambios efectuados implican en el desempeño del modelo y la compensación al modificar uno u otro parámetro. Se implementaron tres tipos de modelos cuyas iteraciones con diferentes configuraciones de hiperparámetros se evaluaron visualmente y se optimizaron en grid search para obtener la mejor configuración correspondiente en cada tipo. Se eligió un GBRcon tasa de aprendizaje = 0.2, profundidad máxima = 7 y número de estimadores = 220; un ABR con tasa de aprendizaje = 0.9, función de pérdida cuadrática y número de estimadores = 150; un SVR con 'C' = 10 , polinomial de grado = 2 con gamma = 0.01. Posteriormente se llevó a cabo una etapa de validación cruzada comparando los 3 modelos elegidos a fin de obtener el mejor.

El desempeño final del modelo GBR fue superlativo y se evaluó con el conjunto restante de prueba, cuyos resultados de validación final presentaron un RMSE de 37,81 y un R2 de 0,99. Adicionalmente, se realizó una revisión de otras métricas de desempeño que involucraron MAE

(23,57), MAPE (0,10) y MSE (37,80). Este desempeño en test hace que el modelo sea bastante robusto y lo suficientemente flexible para adaptarse a más datos explicando la variabilidad de forma adecuada. El desempeño superior del modelo depende netamente de las circunstancias de los datos y se extrajo su configuración a las pruebas previas reportadas.

El modelo elegido proporcionó más peso a las variables: vel_max, distancia, vel_prom y ACCELERATION_Z. Si bien existen más variables, estas le permiten discernir al momento de elegir la medida del RUL de manera más eficiente, por lo que pueden convertirse en un insumo de importancia para futuros trabajos en los que se quiera profundizar en algunas variables específicas e incluso apoyar la toma de decisiones para los fabricantes en cuanto a qué tipo de sensores y en qué áreas ubicarlos y para el grupo de investigación que lleva a cabo la experimentación, para optimizar los recursos usados en el proceso.

11. Recomendaciones

A futuro es muy conveniente explorar diferentes estrategias de recolección de datos, en entornos de mayor control, que permitan un conjunto de variables experimentales limpias, reduciendo los pasos de limpieza y aumentando la cantidad de registros y ciclos disponible.

También es importante tener en cuenta que este enfoque del RUL es la confiabilidad y su interpretabilidad está ligada al tiempo hasta la descarga, lo que puede modificarse con estrategias que exploren los resultados por ciclo, implicando la recolección de una mayor cantidad de ciclos.

Referencias

Ansari, S., Ayob, A., Hossain Lipu, M. S., Hussain, A., & Saad, M. H. M. (2022). Remaining useful life prediction for lithium-ion battery storage system: A comprehensive review of methods, key factors, issues and future outlook. *Energy Reports*, *8*, 12153-12185. https://doi.org/10.1016/j.egyr.2022.09.043

Bracale, A., De Falco, P., Di Noia, L. P., & Rizzo, R. (2023). Probabilistic State of Health and Remaining Useful Life Prediction for Li-Ion Batteries. *IEEE Transactions on Industry Applications*, *59*(1), 578-590. https://doi.org/10.1109/TIA.2022.3210081

Carvalho, T. M. N., & de Assis de Souza Filho, F. (2021). Variational Mode Decomposition Hybridized With Gradient Boost Regression for Seasonal Forecast of Residential Water Demand. *Water Resources Management*, *35*(10), 3431-3445. https://doi.org/10.1007/s11269-021-02902-7

Chen, K., Liao, Q., Liu, K., Yang, Y., Gao, G., & Wu, G. (2023). Capacity degradation prediction of lithium-ion battery based on artificial bee colony and multi-kernel support vector regression. *Journal of Energy Storage*, *72*, 108160. https://doi.org/10.1016/j.est.2023.108160

Datta, P., Das, P., & Kumar, A. (2022). Hyper parameter tuning based gradient boosting algorithm for detection of diabetic retinopathy: an analytical review. *Bulletin of Electrical Engineering and Informatics*, *11*(2), 814-824. https://doi.org/10.11591/eei.v11i2.3559

de Myttenaere, A., Golden, B., Le Grand, B., & Rossi, F. (2016). Mean Absolute Percentage Error for regression models. *Neurocomputing*, *192*, 38-48. https://doi.org/10.1016/j.neucom.2015.12.114

Ding, N., Prasad, K., & Lie, T. T. (2017). The electric vehicle: a review. En *Int. J. Electric and Hybrid Vehicles* (Vol. 9, Número 1).

Dodge Yadola. (2008). Coefficient of Determination. En *The Concise Encyclopedia of Statistics* (pp. 88-91). Springer New York. https://doi.org/10.1007/978-0-387-32833-1_62

Ekanayake, I. U., Palitha, S., Gamage, S., Meddage, D. P. P., Wijesooriya, K., & Mohotti, D. (2023). Predicting adhesion strength of micropatterned surfaces using gradient boosting models and explainable artificial intelligence visualizations. *Materials Today Communications*, *36*, 106545. https://doi.org/10.1016/j.mtcomm.2023.106545

Harper, G., Sommerville, R., Kendrick, E., Driscoll, L., Slater, P., Stolkin, R., Walton, A., Christensen, P., Heidrich, O., Lambert, S., Abbott, A., Ryder, K., Gaines, L., & Anderson, P.

(2019). Recycling lithium-ion batteries from electric vehicles. En *Nature* (Vol. 575, Número 7781, pp. 75-86). Nature Publishing Group. https://doi.org/10.1038/s41586-019-1682-5

Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development*, *15*(14), 5481-5487. https://doi.org/10.5194/gmd-15-5481-2022

Jordaan, E. M., & Smits, G. F. (s. f.). Estimation of the regularization parameter for support vector regression. *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, 2192-2197. https://doi.org/10.1109/IJCNN.2002.1007481

Laeli, A. R., Rustam, Z., Hartini, S., Maulidina, F., & Aurelia, J. E. (2020a). Hyperparameter Optimization on Support Vector Machine using Grid Search for Classifying Thalassemia Data. *2020 International Conference on Decision Aid Sciences and Application (DASA)*, 817-821. https://doi.org/10.1109/DASA51403.2020.9317227

Laeli, A. R., Rustam, Z., Hartini, S., Maulidina, F., & Aurelia, J. E. (2020b). Hyperparameter Optimization on Support Vector Machine using Grid Search for Classifying Thalassemia Data. *2020 International Conference on Decision Aid Sciences and Application (DASA)*, 817-821. https://doi.org/10.1109/DASA51403.2020.9317227

Mathew, J., Luo, M., & Pang, C. K. (2017). Regression kernel for prognostics with support vector machines. *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1-5. https://doi.org/10.1109/ETFA.2017.8247740

Pang, X., Yang, W., Wang, C., Fan, H., Wang, L., Li, J., Zhong, S., Zheng, W., Zou, H., Chen, S., & Liu, Q. (2023). A novel hybrid model for lithium-ion batteries lifespan prediction with high accuracy and interpretability. *Journal of Energy Storage*, *61*. https://doi.org/10.1016/j.est.2023.106728

Patil, M. A., Tagade, P., Hariharan, K. S., Kolake, S. M., Song, T., Yeo, T., & Doo, S. (2015). A novel multistage Support Vector Machine based approach for Li ion battery remaining useful life estimation. *Applied Energy*, *159*, 285-297. https://doi.org/10.1016/j.apenergy.2015.08.119

Raschka, S., Patterson, J., & Nolet, C. (2020). Machine Learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *arXiv preprint arXiv:2002.04803*.

Rochim, A. F., Widyaningrum, K., & Eridani, D. (2021). Performance Comparison of Support Vector Machine Kernel Functions in Classifying COVID-19 Sentiment. *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 224-228. https://doi.org/10.1109/ISRITI54043.2021.9702845

Roy, A., & Chakraborty, S. (2023). Support vector machine in structural reliability analysis: A review. *Reliability Engineering & System Safety*, *233*, 109126. https://doi.org/10.1016/j.ress.2023.109126

Sanguesa, J. A., Torres-Sanz, V., Garrido, P., Martinez, F. J., & Marquez-Barja, J. M. (2021). A review on electric vehicles: Technologies and challenges. En *Smart Cities* (Vol. 4, Número 1, pp. 372-404). MDPI. https://doi.org/10.3390/smartcities4010022

Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*, 526-534. https://doi.org/10.1016/j.procs.2021.01.199

Solomatine, D. P., & Shrestha, D. L. (s. f.). AdaBoost.RT: a boosting algorithm for regression problems. *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, 1163-1168. https://doi.org/10.1109/IJCNN.2004.1380102

Yuan, J., Qin, Z., Huang, H., Gan, X., Li, S., & Li, B. (2023). State of Health Estimation and Remaining Useful Life Prediction for a Lithium-Ion Battery with a Two-Layer Stacking Regressor. *Energies*, *16*(5). https://doi.org/10.3390/en16052313

Zhang, Y., & Zhao, M. (2023). Cloud-based in-situ battery life prediction and classification using machine learning. *Energy Storage Materials*, *57*, 346-359. https://doi.org/10.1016/j.ensm.2023.02.035

Zhao, J., Ling, H., Liu, J., Wang, J., Burke, A. F., & Lian, Y. (2023). Machine learning for predicting battery capacity for electric vehicles. *eTransportation*, *15*. https://doi.org/10.1016/j.etran.2022.100214