Modelado del Consumo Energético de Vehículos mediante Aprendizaje Automático

Valentino N. Fadel
Ingeniería en Inteligencia Artificial
Universidad de San Andrés
Buenos Aires, Argentina
vfadel@udesa.edu.ar

Joaquín Di Cola
Ingeniería en Inteligencia Artificial
Universidad de San Andrés
Buenos Aires, Argentina
jdicola@udesa.edu.ar

***Resumen*—En este trabajo abordamos el problema de modelar el consumo energético de vehículos livianos utilizando el *Vehicle Energy Dataset* (VED). Planteamos el problema como una tarea de regresión supervisada con dos variables objetivo: consumo de combustible en $\mathrm{L}/100\mathrm{km}$ para vehículos con motor de combustión (ICE, HEV, PHEV) y consumo eléctrico en kWh/km para vehículos con batería de alta tensión (PHEV, EV).**

Construimos dos representaciones de las señales dinámicas del vehículo: un conjunto de 58 métricas estadísticas en el dominio del tiempo (*Metrics*) y 86 descriptores frecuenciales basados en la transformada rápida de Fourier (*Fourier*). Sobre ambas representaciones entrenamos y comparamos modelos de regresión basados en árboles (Random Forest, XGBoost y LightGBM) y modelos lineales como línea de base.

Los resultados muestran que la representación *Metrics* supera a *Fourier* con $R^2 = 0{,}68$ frente a $0{,}47$. El modelo final XGBoost, optimizado mediante RandomizedSearchCV con 50 iteraciones y validación cruzada de 5 folds, alcanza un $R^2 = 0{,}80$ y RMSE de 2,26 $\mathrm{L}/100\mathrm{km}$ para combustión, y $R^2 = 0{,}63$ con RMSE de 0,069 kWh/km para consumo eléctrico.

I. Introducción

El transporte automotor representa una fracción significativa del consumo global de energía y de las emisiones de gases de efecto invernadero. Mejorar la eficiencia energética de los vehículos resulta fundamental tanto desde el punto de vista ambiental como económico. En este contexto, la disponibilidad de datos masivos provenientes de sensores a bordo habilita el uso de modelos de aprendizaje automático para entender cómo las condiciones de operación y el estilo de conducción influyen sobre el consumo.

En este proyecto utilizamos el *Vehicle Energy Dataset* (VED), un conjunto de datos desarrollado por el U.S. Department of Energy que recopila mediciones reales de 383 vehículos livianos de distintos tipos de motorización: 264 vehículos a gasolina (ICE), 92 híbridos (HEV) y 27 híbridos enchufables o eléctricos puros (PHEV/EV). Cada vehículo fue instrumentado con sensores OBD-II y GPS, registrando velocidad, aceleración, temperatura ambiente, potencias auxiliares y estado de la batería.

Nuestro objetivo es predecir el consumo energético promedio de cada trayecto a partir de variables dinámicas del vehículo y del entorno, considerando tanto el consumo de combustible como el consumo eléctrico según el tipo de vehículo.

II. Conjunto de datos y características

II-A. *Vehicle Energy Dataset*

El VED combina dos tipos de información. Por un lado, los datos estáticos incluyen características del vehículo como tipo de motor, transmisión, peso y tipo de tracción. Por otro lado, los datos dinámicos corresponden a mediciones registradas cada segundo durante la conducción, tales como velocidad del vehículo, aceleración, temperatura exterior, potencias auxiliares, estado de carga de la batería de alta tensión y variables relacionadas al sistema de combustible.

En total procesamos 32.512 trayectos, cada uno identificado por una combinación única de archivo, identificador de vehículo, día y número de viaje. Para evitar *data leakage*, las variables empleadas para calcular el consumo (Fuel Rate, MAF, HV Battery Current/Voltage/SOC, Fuel Trims) se removieron del conjunto de features.

II-B. *Cálculo de targets*

Definimos dos variables objetivo según el tipo de vehículo. El consumo de combustión, expresado en $\mathrm{L}/100\mathrm{km}$, aplica a vehículos ICE, HEV y PHEV, y se obtuvo integrando el flujo de combustible a lo largo del tiempo y normalizándolo por la distancia recorrida, resultando en un dataset de 26.633 muestras. El consumo eléctrico, expresado en kWh/km, aplica a vehículos PHEV y EV, y se calculó a partir de la potencia de la batería de alta tensión (corriente multiplicada por voltaje) integrada en el tiempo, resultando en 4.124 muestras.

II-C. *Representaciones de features*

Se consideraron dos representaciones de las señales dinámicas. La representación *Metrics* consiste en 58 features donde para cada señal relevante se calcularon estadísticas en el dominio del tiempo, incluyendo media, mediana, desvío estándar, mínimos, máximos, percentiles 25/75 y rango. La representación *Fourier* comprende 86 features obtenidas mediante la transformada rápida de Fourier (FFT) a nivel de trayecto, extrayendo las magnitudes y frecuencias de los primeros armónicos, así como la energía espectral y el centroide del espectro.

II-D. Partición de datos y normalización

Los trayectos se dividieron en tres conjuntos mutuamente excluyentes: train con aproximadamente el 64 % de las muestras, validation con el 16 %, y test con el 20 % restante. Sobre el conjunto de train se ajustó un *StandardScaler*, que luego se aplicó a validation y test. Los valores faltantes se imputaron con la media de cada feature.

III. Metodología

III-A. Modelos de regresión

Se evaluaron seis modelos de regresión, divididos en dos categorías: modelos basados en árboles y modelos lineales.

Random Forest Regressor es un método de ensamble que construye B árboles de decisión independientes, cada uno entrenado sobre una muestra bootstrap del conjunto de entrenamiento. La predicción final es el promedio de las predicciones individuales:

$$\hat{y} = \frac{1}{B}\sum_{b=1}^{B} T_b(\mathbf{x}) \tag{1}$$

donde $T_b(\mathbf{x})$ es la predicción del árbol b. En cada nodo se selecciona aleatoriamente un subconjunto de m features (típicamente $m = \sqrt{p}$ para clasificación o $m = p/3$ para regresión), reduciendo la correlación entre árboles y mejorando la generalización.

XGBoost Regressor implementa *gradient boosting* construyendo árboles secuencialmente. En la iteración t, el modelo acumulado es $F_t(\mathbf{x}) = F_{t-1}(\mathbf{x}) + \eta \cdot h_t(\mathbf{x})$, donde η es la tasa de aprendizaje y h_t minimiza:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{N} \ell(y_i, \hat{y}_i^{(t-1)} + h_t(\mathbf{x}_i)) + \Omega(h_t) \tag{2}$$

La regularización $\Omega(h_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 + \alpha \sum_{j=1}^{T} |w_j|$ penaliza el número de hojas T y los pesos w_j, previniendo sobreajuste mediante términos L1 (α) y L2 (λ).

LightGBM Regressor comparte la formulación de gradient boosting pero difiere en la estrategia de crecimiento: utiliza expansión *leaf-wise* (siempre divide la hoja con mayor ganancia) en lugar de *level-wise*, produciendo árboles asimétricos más profundos. La ganancia de una división se calcula como:

$$\text{Gain} = \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda}\right] - \gamma \tag{3}$$

donde G y H son las sumas de gradientes y hessianos de las muestras en cada partición.

Modelos lineales se incluyeron como línea base. Todos buscan coeficientes $\boldsymbol{\beta}$ que minimicen una función de pérdida. OLS minimiza $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$. Ridge añade penalización L2:

$$\hat{\boldsymbol{\beta}}_{\text{Ridge}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_2^2 \tag{4}$$

Lasso utiliza penalización L1: $\hat{\boldsymbol{\beta}}_{\text{Lasso}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1$, lo que puede reducir coeficientes exactamente a cero, realizando selección de variables.

III-B. Optimización de hiperparámetros

Para el modelo final se utilizó RandomizedSearchCV con 50 iteraciones y validación cruzada de 5 folds (*k-fold cross-validation*). Esta técnica divide el conjunto de entrenamiento en 5 particiones disjuntas; en cada iteración, 4 particiones se usan para entrenar y 1 para validar, rotando hasta que todas hayan servido como validación. El desempeño final es el promedio de las 5 evaluaciones, lo que proporciona una estimación más robusta y reduce la varianza respecto a una única partición train/validation.

El espacio de búsqueda de hiperparámetros para XGBoost incluyó: n_estimators entre 100 y 500, max_depth entre 3 y 12, learning_rate entre 0,01 y 0,30, subsample entre 0,6 y 1,0, colsample_bytree entre 0,6 y 1,0, min_child_weight entre 1 y 10, gamma entre 0 y 0,5, reg_alpha entre 0 y 1, y reg_lambda entre 0 y 2.

IV. Experimentos, resultados y discusión

IV-A. Métricas de evaluación

Utilizamos como métricas principales el error cuadrático medio (RMSE), el error absoluto medio (MAE) y el coeficiente de determinación R^2:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \tag{5}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|, \tag{6}$$

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}. \tag{7}$$

IV-B. Comparación de modelos sobre representación Metrics

La Tabla I resume el desempeño de los modelos sobre features *Metrics* en el conjunto de validation para el target de combustión. Random Forest obtiene el mejor desempeño con $R^2 = 0{,}86$ y RMSE de 1,91 $\text{L}/100\text{km}$, seguido por XGBoost con $R^2 = 0{,}79$. Los modelos lineales alcanzan un R^2 de apenas 0,43, evidenciando que la relación entre features y consumo es fuertemente no lineal.

Los modelos lineales (OLS, Ridge, Lasso) fracasan porque asumen una relación lineal $y = \mathbf{X}\boldsymbol{\beta} + \epsilon$ entre features y target, incapaz de capturar las interacciones complejas entre variables como velocidad, régimen del motor y condiciones ambientales. Por ejemplo, el consumo no crece linealmente con la velocidad: es alto a bajas velocidades (arranques frecuentes), decrece a velocidades moderadas (eficiencia óptima), y vuelve a aumentar a altas velocidades (resistencia aeronámica). Esta relación no monótona es imposible de modelar con combinaciones lineales de features.

En contraste, los modelos basados en árboles triunfan porque pueden capturar interacciones de alto orden y relaciones no lineales mediante particiones jerárquicas del espacio de features. Random Forest reduce la varianza promediando múltiples árboles descorrelacionados, mientras que XGBoost

y LightGBM corrigen iterativamente los errores residuales, logrando menor sesgo.

Cuadro I
MODELOS SOBRE FEATURES *Metrics* - COMBUSTIÓN (VALIDATION).

Modelo	RMSE	MAE	R^2
Random Forest	1,91	1,18	0,86
XGBoost	2,32	1,16	0,79
LightGBM	2,42	1,22	0,77
Linear OLS	3,79	2,03	0,43
Ridge	3,78	2,03	0,43
Lasso	3,78	2,03	0,43

IV-C. Comparación de representaciones: Metrics vs Fourier

Para features *Fourier*, XGBoost lidera con $R^2 = 0{,}66$, mientras que Random Forest cae a $R^2 = 0{,}45$. Esta inversión sugiere que los descriptores frecuenciales requieren modelos con mayor capacidad de regularización para evitar sobreajuste. Random Forest falla con features Fourier porque sus 86 descriptores espectrales presentan alta dimensionalidad y correlación entre armónicos, lo que genera árboles con particiones ruidosas. XGBoost, al construir árboles secuencialmente y aplicar regularización L1/L2, filtra mejor las features irrelevantes.

La Figura 1 presenta la comparación directa entre ambas representaciones evaluadas sobre el conjunto de test. Se observa que *Metrics* supera consistentemente a *Fourier* en las tres métricas: MSE de 8,31 frente a 13,53, MAE de 1,78 frente a 2,45, y R^2 de 0,68 frente a 0,47. Esta diferencia de 21 puntos porcentuales en R^2 indica que las estadísticas en el dominio del tiempo capturan mejor la información relevante para predecir el consumo.

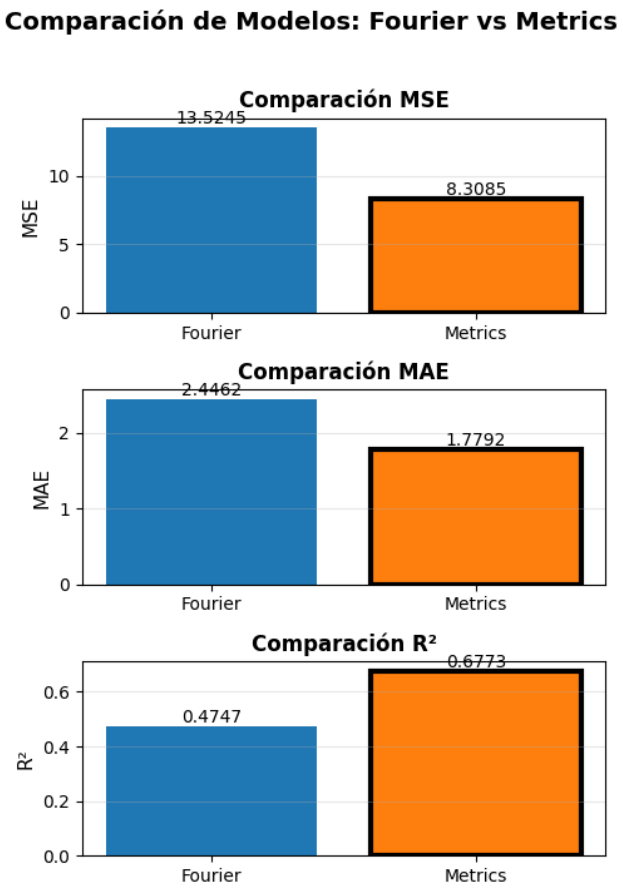

Figura 1. Comparación de MSE, MAE y R^2 entre representaciones Metrics y Fourier sobre el conjunto de test.

El análisis sugiere que la transformada de Fourier, si bien captura patrones periódicos de conducción, pierde información sobre la magnitud absoluta de las señales, la cual resulta crítica para estimar el consumo. Las métricas estadísticas preservan tanto la escala como la variabilidad de las señales originales. Además, la FFT asume estacionariedad de las señales, lo cual no se cumple en trayectos vehiculares donde las condiciones de conducción varían continuamente (arranques, frenadas, velocidad crucero).

IV-D. Modelo final: XGBoost optimizado

Dado que *Metrics* resultó la mejor representación, se procedió a entrenar modelos finales XGBoost optimizados para cada target mediante RandomizedSearchCV con 50 iteraciones y validación cruzada de 5 folds. Se seleccionó XGBoost por su equilibrio entre rendimiento y capacidad de generalización, así como por su robustez frente al sobreajuste gracias a la regularización L1/L2.

La Tabla II presenta los resultados finales sobre el conjunto de test. El modelo de combustión alcanza un $R^2 = 0{,}80$ con RMSE de 2,26 $\mathrm{L}/100\mathrm{km}$ sobre 5.327 muestras, representando una mejora del 55 % en RMSE respecto al baseline. El modelo eléctrico obtiene un $R^2 = 0{,}63$ con RMSE de 0,069 kWh/km sobre 825 muestras, con una mejora del 39 %.

Cuadro II
RESULTADOS FINALES EN EL CONJUNTO DE TEST.

Target	Muestras	RMSE	MAE	R^2
Combustión (L/100km)	5.327	2,26	1,17	0,80
Eléctrico (kWh/km)	825	0,069	0,035	0,63

IV-E. Evaluación detallada de modelos finales

La Figura 2 muestra la evaluación del modelo de combustión mediante cuatro visualizaciones. El gráfico de predicción versus valor real muestra dispersión uniforme alrededor de la diagonal, indicando buena calibración. Los residuos en función de la predicción tienen media cercana a cero, indicando ausencia de sesgo sistemático, aunque se observa heterocedasticidad: la varianza aumenta con el valor predicho. La distribución de residuos se aproxima a una normal centrada en cero.

A pesar de su buen desempeño global, el modelo de combustión presenta limitaciones. La heterocedasticidad observada indica que el modelo falla más en trayectos de alto consumo (conducción agresiva, tráfico intenso), donde la variabilidad inherente es mayor. Además, los outliers con errores superiores a 5 $\mathrm{L}/100\mathrm{km}$ corresponden típicamente a condiciones extremas no bien representadas en el entrenamiento: trayectos muy cortos con múltiples arranques en frío, o condiciones climáticas atípicas.

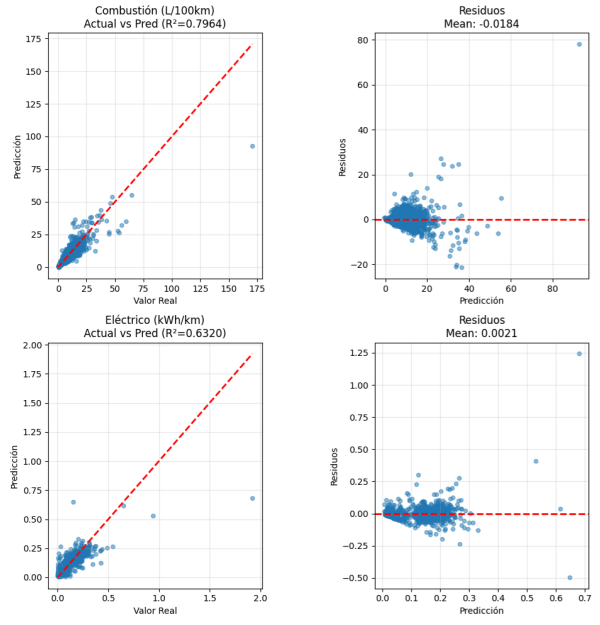

Figura 2. Evaluación del modelo de combustión ($\mathrm{L}/100\mathrm{km}$). De izquierda a derecha: predicción vs. valor real, residuos vs. predicción, distribución de residuos, y top 10 features.

La Figura 3 presenta el análisis equivalente para el modelo eléctrico. Este modelo muestra mayor variabilidad en consumos bajos, lo cual se explica por la menor cantidad de datos disponibles (825 muestras de test frente a 5.327 para combustión). A pesar de esto, los residuos mantienen distribución aproximadamente normal.

El modelo eléctrico ($R^2 = 0{,}63$) tiene menor rendimiento que el de combustión ($R^2 = 0{,}80$) por varias razones. Primero, el dataset de vehículos eléctricos es 6 veces más pequeño, limitando la capacidad del modelo para aprender patrones complejos. Segundo, el consumo eléctrico depende fuertemente de la regeneración durante frenadas, un fenómeno difícil de capturar con estadísticas agregadas por trayecto. Tercero, los vehículos PHEV pueden alternar entre modos eléctrico y combustión durante un mismo trayecto, introduciendo variabilidad no explicada por las features disponibles.

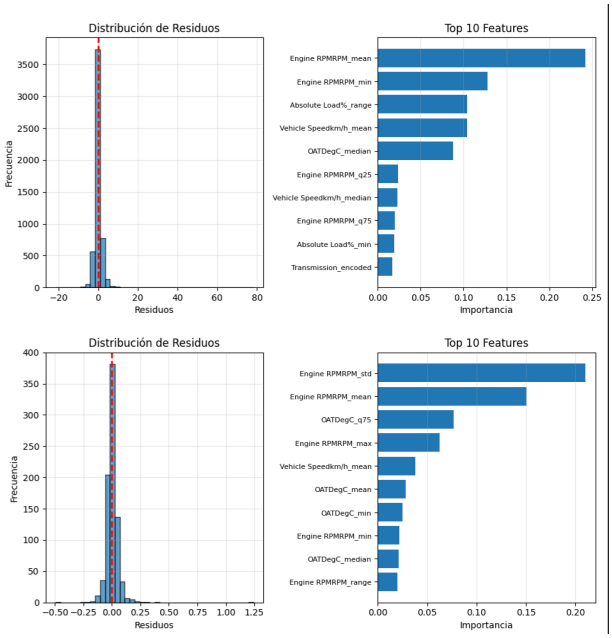

Figura 3. Evaluación del modelo eléctrico (kWh/km). De izquierda a derecha: predicción vs. valor real, residuos vs. predicción, distribución de residuos, y top 10 features.

IV-F. Importancia de variables

Las Figuras 2 y 3 muestran en su cuarta columna las 10 features más importantes para cada modelo. Las Figuras 4 y 5 amplían este análisis mostrando las 15 features principales.

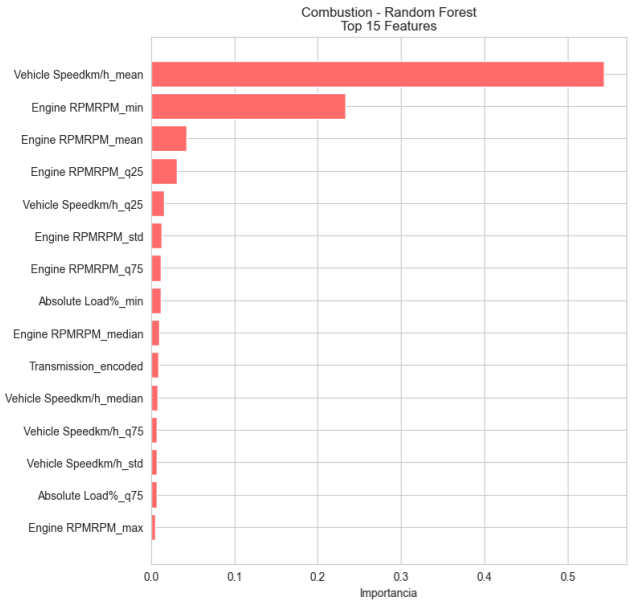

Figura 4. Top 15 features por importancia para el modelo de combustión.

Las variables más relevantes para combustión incluyen estadísticas de velocidad del vehículo (media, percentiles, rango), régimen del motor (Engine RPM), potencias de climatización y temperatura exterior. Estos resultados son consistentes con la intuición física: mayor régimen implica mayor consumo, y los sistemas auxiliares demandan energía adicional.

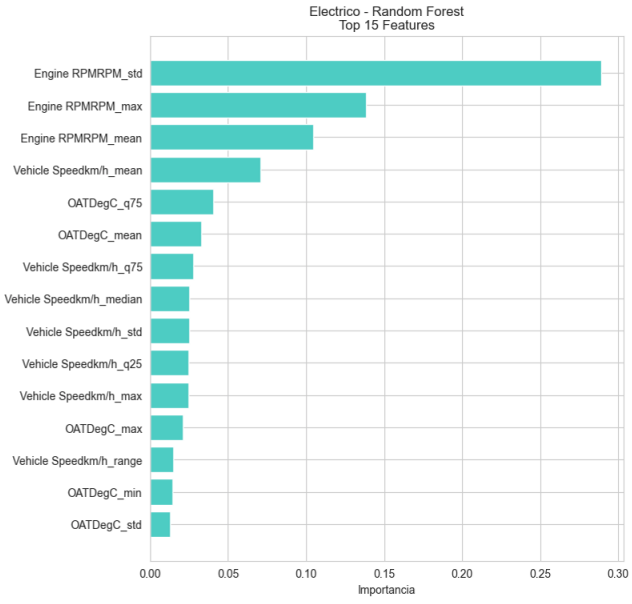

Figura 5. Top 15 features por importancia para el modelo eléctrico.

Para el modelo eléctrico, las features más importantes son estadísticas de velocidad y potencias auxiliares, con menor peso de variables del motor de combustión, lo cual es esperable dado que estos vehículos dependen exclusivamente de la batería.

IV-G. Análisis de sobreajuste

Para evaluar el sobreajuste comparamos el desempeño en train y test. El modelo de combustión obtiene $R^2 = 0{,}87$ en train frente a $R^2 = 0{,}80$ en test, indicando una diferencia de solo 7 puntos porcentuales. Esta brecha moderada se mitiga mediante la regularización L1/L2 de XGBoost y la validación cruzada de 5 folds. La partición estricta train/validation/test garantiza que las métricas reportadas reflejan la capacidad de generalización real.

V. Conclusiones y trabajo futuro

En este proyecto desarrollamos modelos de regresión para estimar el consumo energético de trayectos vehiculares utilizando el *Vehicle Energy Dataset*. Los modelos basados en árboles superan ampliamente a los modelos lineales, confirmando que la relación entre las características del trayecto y el consumo es inherentemente no lineal.

La representación *Metrics*, con 58 features estadísticas en el dominio del tiempo, supera a *Fourier* con 86 features frecuenciales, alcanzando un R^2 de 0,68 frente a 0,47 en el conjunto de test. Este resultado sugiere que para la tarea de predicción de consumo energético, las estadísticas descriptivas tradicionales capturan mejor la información relevante que los descriptores espectrales.

El modelo final XGBoost optimizado mediante búsqueda de hiperparámetros alcanza un R^2 de 0,80 para combustión y 0,63 para consumo eléctrico. La diferencia de rendimiento entre ambos modelos se explica principalmente por el desbalance en la cantidad de datos: 26.633 muestras para combustión frente a solo 4.124 para eléctrico.

Como trabajo futuro se propone incorporar características estáticas del vehículo, como peso y tipo de motor, para mejorar la capacidad de generalización entre vehículos heterogéneos. Finalmente, recopilar más datos de vehículos eléctricos permitiría mejorar sustancialmente el modelo de consumo eléctrico.

Referencias

[1] G. Oh, D. J. LeBlanc, and H. Peng, "Vehicle Energy Dataset (VED), A Large-scale Dataset for Vehicle Energy Consumption Research," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 3302–3312, 2022. Disponible en: https://github.com/gsoh/VED. Dataset utilizado bajo licencia CC BY 4.0.

[2] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[3] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[4] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD Int. Conf.*, 2016, pp. 785–794.

[5] G. Ke et al., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems*, 2017, pp. 3146–3154.

[6] W. McKinney, "Data Structures for Statistical Computing in Python," in *Proc. 9th Python in Science Conf.*, 2010, pp. 56–61.

[7] C. R. Harris et al., "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, 2020.