DESARROLLO DE UN MODELO DE ANÁLISIS DE CARGA NO INTRUSIVA PARA EFICIENCIA ENERGÉTICA BASADO EN TÉCNICAS DE MACHINE LEARNING

Meidy Lizeth Moreno Guzmán
José Luis Munévar Diaz
Julián Javier Jaimes Otero

Proyecto Aplicado para optar al título de
Magister en Ciencia de Datos

Director(a)
David Arango Londoño

FACULTAD DE INGENIERÍA Y CIENCIAS
MAESTRÍA EN CIENCIA DE DATOS
SANTIAGO DE CALI, JUNIO 30 DE 2026

FICHA RESUMEN

TÍTULO DEL PROYECTO: Desarrollo De Un Modelo De Análisis De Carga No Intrusiva Para Eficiencia Energética Basado En Técnicas De Machine Learning.

1. ÁREA DE TRABAJO: Eficiencia Energética.
2. TIPO DE PROYECTO: Aplicado.
3. ESTUDIANTE(S):

   Meidy Lizeth Moreno, José Luis Munévar Diaz, Julián Javier Jaimes Otero
4. CORREO ELECTRÓNICO:

   morenoguzm1@javerianacali.edu.co, joseluis281@javerianacali.edu.co, jotajaimes@javerianacali.edu.co.
5. DIRECCIÓN Y TELÉFONO: Cra 17 98-03 Bucaramanga – 3118218420.
6. DIRECTOR: David Arango
7. VINCULACIÓN DEL DIRECTOR: Pontificia Universidad Javeriana Cali, Docente investigador, Departamento de Ciencias de la Computación.
8. CORREO ELECTRÓNICO DEL DIRECTOR: david.arango@javerianacali.edu.co.
9. CO-DIRECTOR (Si aplica): NA.
10. GRUPO O EMPRESA QUE LO AVALA (Si aplica): NA.
11. OTROS GRUPOS O EMPRESAS: NA.
12. PALABRAS CLAVE (al menos 5): Carga No Intrusiva, Eficiencia Energética, Machine Learning, Desagregación de energía, Análisis de series temporales.
13. FECHA DE INICIO: junio 1 de 2025
14. DURACIÓN ESTIMADA (En meses): 12

RESUMEN

Esta investigación desarrolló un sistema de monitoreo de carga no intrusivo (Non-Intrusive Load Monitoring, NILM) basado en técnicas de inteligencia artificial (IA), orientado a mejorar la eficiencia energética en entornos comerciales. La creciente demanda energética, sumada al aumento de dispositivos eléctricos conectados a la red, ha generado un consumo poco eficiente debido, en gran parte, a la falta de información detallada sobre el comportamiento energético de los equipos eléctricos. Los sistemas tradicionales de medición únicamente reportan el consumo total, sin permitir identificar qué equipos consumen más energía, en qué momentos y bajo qué condiciones, lo que dificulta la implementación de estrategias de ahorro y sostenibilidad.

Como alternativa a los sistemas de monitoreo intrusivo, que requieren la instalación de sensores en cada dispositivo y presentan altos costos de implementación y mantenimiento, se desarrolló un modelo NILM capaz de desagregar el consumo energético a partir de una única medición realizada en el punto de entrada de la instalación eléctrica. Para ello, se realizó el procesamiento y análisis exploratorio de un conjunto de datos de consumo energético, se evaluaron cinco modelos de aprendizaje automático (Regresión Lineal, SVM, Random Forest, XGBoost y RNN-LSTM), optimizados mediante técnicas de búsqueda de hiperparámetros, y se implementó un mecanismo de reconciliación proporcional para garantizar la coherencia entre las predicciones desagregadas y el consumo total observado.

Los resultados evidenciaron que los modelos basados en métodos de ensamble, especialmente Random Forest, alcanzaron los mejores niveles de desempeño, obteniendo coeficientes de determinación (R^2) superiores al 90 % en cuatro de las seis categorías de carga evaluadas (Lighting, Heating, Sockets Plug y Cooling). Asimismo, el mecanismo de reconciliación permitió corregir inconsistencias entre las predicciones individuales y el consumo agregado, garantizando la conservación de la energía durante el proceso de desagregación.

En conjunto, los resultados demuestran la viabilidad del uso de técnicas de Machine Learning para la desagregación no intrusiva del consumo energético en edificios comerciales y proporcionan una metodología reproducible que puede apoyar la implementación de sistemas inteligentes de gestión energética, facilitar la toma de decisiones orientadas al ahorro de energía y contribuir al desarrollo de estrategias de sostenibilidad en organizaciones.

Palabras clave: Monitoreo de carga no intrusivo (NILM), Desagregación energética, Aprendizaje automático, Bosques aleatorios (Random Forest), Series temporales, Redes neuronales recurrentes (RNN), Reconciliación de pronósticos.

ABSTRACT

Energy efficiency has become a critical challenge for commercial organizations seeking to reduce operational costs and environmental impact. Traditional energy metering systems provide only aggregated consumption data, making it difficult to identify which electrical loads are responsible for high energy use. Non-Intrusive Load Monitoring (NILM) offers a cost-effective solution by disaggregating total energy consumption into individual load categories from a single measurement point without requiring physical sensors on each device.

This study developed a machine learning-based NILM model to disaggregate energy consumption across six load categories (Lighting, Heating, Sockets Plug, Cooling, Ventilation, and Other Electricity) in a commercial building. The methodology was evaluated using a dataset containing 43,824 hourly energy consumption records. Exploratory data analysis (EDA) was conducted, including outlier detection and temporal interpolation. Five regression models (Linear Regression, SVM, Random Forest, XGBoost, and RNN-LSTM) were evaluated and optimized using Grid Search or RandomizedSearchCV. In addition, a proportional reconciliation mechanism was implemented to ensure consistency between the disaggregated predictions and the total observed energy consumption.

The experimental results demonstrated that Random Forest achieved R^2 test values above 90% for four of the six load categories (Lighting 96.5%, Heating 96.1%, Sockets Plug 92.2%, and Cooling 91.1%), demonstrating the effectiveness of ensemble methods for energy disaggregation. Furthermore, the reconciliation mechanism successfully corrected prediction inconsistencies while preserving energy conservation. These findings provide a reproducible and scalable methodology for implementing non-intrusive energy monitoring systems in commercial buildings and support future research on intelligent energy management using machine learning techniques.

Keywords: Non-Intrusive Load Monitoring (NILM), Energy Disaggregation, Machine Learning, Random Forest, Time Series, Recurrent Neural Networks (RNN), Forecast Reconciliation.

LISTA DE ACRÓNIMOS

Acrónimo	Término Original	Descripción
R^2	-	Coeficiente de Determinación
ACM	Association for Computing Machinery	Asociación de Maquinaria de Computación
BBVA	-	Banco Bilbao Vizcaya Argentaria
CNN	Convolutional Neural Network	Red Neuronal Convolucional
CNN-LSTM	Convolutional Neural Network - Long Short-Term Memory	Modelo híbrido de Red Neuronal Convolucional y Memoria a Largo Corto Plazo
DNN	Deep Neural Network	Red Neuronal Profunda
EDA	Exploratory Data Analysis	Análisis Exploratorio de Datos
FHMM	Factorial Hidden Markov Model	Modelo Factorial Oculto de Markov
HMM	Hidden Markov Model	Modelo Oculto de Markov
IA	-	Inteligencia Artificial
IEA	International Energy Agency	Agencia Internacional de la Energía
IEEE	Institute of Electrical and Electronics Engineers	Instituto de Ingenieros Eléctricos y Electrónicos
ILM	Intrusive Load Monitoring	Monitoreo de Carga Intrusivo
LSTM	Long Short-Term Memory	Memoria a Largo Corto Plazo
MAE	Mean Absolute Error	Error Absoluto Medio
MSE	Mean Squared Error	Error Cuadrático Medio
NILM	Non-Intrusive Load Monitoring	Monitoreo de Carga No Intrusivo / Desagregación de Energía
NN	Neural Network	Red Neuronal
PIB	-	Producto Interno Bruto
PMLR	Proceedings of Machine Learning Research	Actas de Investigación en Aprendizaje Automático
RBF	Radial Basis Function	Función de Base Radial
REDD	Reference Energy Disaggregation Data	Conjunto de Datos de Referencia para Desagregación de Energía
REFIT	Personalised Retrofit Advice Modelling Using Smart Meter Data	Conjunto de Datos Británico de Consumo Eléctrico Residencial
RF	Random Forest	Bosque Aleatorio
RMSE	Root Mean Squared Error	Raíz del Error Cuadrático Medio
RNN	Recurrent Neural Network	Red Neuronal Recurrente
RNN-LSTM	Recurrent Neural Network - Long Short-Term Memory	Red Neuronal Recurrente basada en Memoria a Largo Corto Plazo
SVM	Support Vector Machine	Máquinas de Vectores de Soporte
UK-DALE	Domestic Appliance Level Electricity dataset	Conjunto de Datos Domésticos de Nivel de Electrodomésticos del Reino Unido

TABLA DE CONTENIDO

Pág.

LISTA DE TABLAS 8

LISTA DE FIGURAS 8

INTRODUCCIÓN 10

1. CONTEXTUALIZACIÓN DEL PROYECTO 11

1.1. Definición del problema 11
1.1.1. Planteamiento del problema 11
1.1.2. Formulación del problema 12
1.1.3. Sistematización 12
1.1.3.1. Procesamiento y recolección de datos 12
1.1.3.2. Evaluar diferentes modelos de machine learning 13
1.1.3.3. Desarrollar un mecanismo de reconciliación. 13
1.1.3.4. Evaluar la precisión y eficiencia del modelo 13
1.1.3.5. Apropiación social del conocimiento y soporte a la toma de decisiones 13

1.2. Objetivos del proyecto 14
1.2.1. Objetivo General 14
1.2.2. Objetivos Específicos 14

1.3. Marco de referencia 15
1.3.1. Marco teórico 15
1.3.1.1. Eficiencia energética y su importancia 15
1.3.1.2. Monitoreo de energía: de lo intrusivo a lo inteligente 16
1.3.1.3. Componentes técnicos del NILM 16
1.3.1.4. Fundamentos de la regresión línea múltiple 17
1.3.1.5. Máquinas de vectores de soporte para regresión (SVR) 17
1.3.1.6. Bosques aleatorios (Random Forest) 18
1.3.1.7. Gradient boosting y XGBoost 19
1.3.1.8. Redes neuronales recurrentes 19
1.3.1.9. Técnicas de inteligencia artificial aplicadas al NILM 20
1.3.1.10. Retos del monitoreo no intrusivo 21
1.3.1.11. Aplicaciones prácticas y beneficios 21
1.3.1.12. Métricas de evaluación del desempeño 22
1.3.2. Reconciliación de pronósticos en series temporales jerárquicas 22
1.3.3. Antecedentes 23

1.4. Estado del arte 24
1.4.1 Métodos clásicos y probabilísticos 24
1.4.2. Métodos basados en aprendizaje profundo 24
1.4.3. Aplicaciones en edificios comerciales con modelos de ensamble 24

2. PROCESAMIENTO Y RECOLECCIÓN DE DATOS DE CARGA ENERGÉTICA Y CONSUMO ELÉCTRICO A TRAVÉS DEL ANÁLISIS EXPLORATORIO Y CORRELACIONES. 26

2.1. Introducción **26**

2.2. Análisis exploratorio y limpieza de datos **28**
2.2.1. Análisis exploratorio (EDA) 28
2.2.2. Limpieza de datos 35

3. EVALUACIÓN DE MODELOS DE MACHINE LEARNING PARA DESAGREGAR EL CONSUMO ENERGÉTICO. 38

3.1 Selección de modelos candidatos **38**

3.2 Preparación y preprocesamiento de datos para el modelado **39**

3.3 Diseño experimental y partición de datos **40**

3.4 Optimización de hiperparámetros **41**

3.5 Flujo metodológico para el entrenamiento y evaluación de modelos **43**

4. RESULTADOS 45

4.1 Resultados de la variable Lighting **45**

4.2 Resultados de la variable Heating **47**

4.3 Resultados de la variable Socket Plug **50**

4.4 Resultados de la variable Cooling **52**

4.5 Resultados de la variable Ventilation **54**

4.6 Resultados de la variable Other electricity **56**

4.7 Mejores resultados de desempeño por variable **58**

4.8 Hiperparámetros óptimos identificados **60**

4.9 Reconciliación de modelos **61**

4.10 Evaluación estadística resultados del error en la reconciliación. **62**

5. CONCLUSIONES 64

6. TRABAJOS FUTUROS 66

7. REFERENCIAS BIBLIOGRÁFICAS 67

8. ANEXOS. 69

LISTA DE TABLAS

Tabla 1. Encabezado del conjunto de datos. 28
Tabla 2. Resumen del conjunto de datos. 29
Tabla 3. Resumen valores atípicos. 30
Tabla 4. Registros inconsistencias temporales. 31
Tabla 5. Cantidad de modelos por referencias 38
Tabla 6. Clasificación de variables 40
Tabla 7. Espacio de hiperparámetros fase de búsqueda 42
Tabla 8. Desempeño de los modelos en la variable Lighting. 46
Tabla 9. Configuraciones de hiperparámetros evaluadas en la variable Lighting. 47
Tabla 10. Desempeño de los modelos en la variable Heating. 48
Tabla 11. Configuraciones de hiperparámetros evaluadas en la variable Heating. 49
Tabla 12. Desempeño de los modelos en la variable Socket Plug. 50
Tabla 13. Configuraciones de hiperparámetros evaluadas en la variable Socket Plug. 51
Tabla 14. Desempeño de los modelos en la variable Cooling. 52
Tabla 15. Configuraciones de hiperparámetros evaluadas en la variable Cooling. 53
Tabla 16. Desempeño de los modelos en la variable Ventilation. 54
Tabla 17. Configuraciones de hiperparámetros evaluadas en la variable Ventilation. 55
Tabla 18. Desempeño de los modelos en la variable Other electricity. 56
Tabla 19. Configuraciones de hiperparámetros evaluadas en la variable Other electricity. 57
Tabla 20. Hiperparámetros asociados a los modelos con mejor desempeño 60
Tabla 21. Resultados de desempeño de modelos de regresión 69

LISTA DE FIGURAS

Imagen 1. Metodología de desagregación y pronóstico energético basada en NILM 26
Imagen 2. Matriz de Correlación de Spearman. 32
Imagen 3. Consumos por día-hora. 33
Imagen 4. Consumos por año-mes. 34
Imagen 5. Distribución de valores atípicos Set de datos original. 36

Imagen 6. Set de datos sin valores atípicos. 37
Imagen 7. Flujo metodológico para el entrenamiento. 44
Imagen 8. Mejor R^2 por variable. 59
Imagen 9. Distribución del error de la reconciliación. 63
Imagen 10. Boxplot del error de la reconciliación. 63

INTRODUCCIÓN

En la era de la revolución tecnológica, las empresas dependen cada vez más de sistemas eléctricos que les permitan mantenerse conectadas, garantizar la seguridad y, en esencia, operar de manera continua. En este contexto, el consumo de energía eléctrica ha crecido de forma exponencial en los últimos años, una tendencia que probablemente continuará debido a esta creciente dependencia. Sin embargo, los sistemas de medición tradicionales instalados en la mayoría de los entornos empresariales proporcionan únicamente datos agregados del consumo total, sin permitir identificar qué equipos o sistemas responsables de los mayores consumos.

La eficiencia energética se ha convertido en una prioridad para las organizaciones que buscan optimizar sus recursos, reducir costos operativos y contribuir a la sostenibilidad ambiental. No obstante, la desagregación del consumo eléctrico para identificar oportunidades de ahorro suele requerir inversiones significativas en infraestructura, como la instalación de medidores individuales en cada dispositivo, lo cual no siempre es viable técnica o económicamente.

En este contexto, las tecnologías de inteligencia artificial y ciencia de datos ofrecen nuevas oportunidades para abordar este desafío. En particular, la desagregación de carga no intrusiva mediante modelos de machine learning permite estimar el consumo eléctrico por tipo de carga (por ejemplo, iluminación, computadores o sistemas de climatización) a partir de datos agregados previamente almacenados. Esto evita la necesidad de instalar sensores en cada equipo y proporciona información detallada que puede ser utilizada para detectar patrones de uso, identificar ineficiencias energéticas y predecir el comportamiento del consumo futuro.

Este proyecto propone el desarrollo de un modelo basado en técnicas de machine learning que permita desagregar el consumo eléctrico por categorías en un edificio de oficinas. El objetivo es generar información que facilite a las empresas una gestión energética más eficiente, mediante predicciones que sustenten la toma de decisiones orientadas a la reducción de costos y al consumo sostenible

Con el propósito de desarrollar esta investigación de manera estructurada, el documento se organiza en cinco capítulos principales. El primero presenta la contextualización del problema, los objetivos, el marco de referencia y el estado del arte. El segundo describe el procesamiento y análisis exploratorio de los datos utilizados en la investigación. El tercero expone la metodología empleada para la evaluación de los modelos de aprendizaje automático y la optimización de sus hiperparámetros. El cuarto presenta los resultados obtenidos, incluyendo el análisis comparativo de los modelos, el proceso de reconciliación y la evaluación de su desempeño. Finalmente, el quinto capítulo reúne las conclusiones de la investigación y los trabajos futuros derivados de este estudio.

1. CONTEXTUALIZACIÓN DEL PROYECTO

1.1. Definición del problema

El desperdicio energético y el envejecimiento o mal estado de los equipos eléctricos representan factores claves en el incremento del consumo innecesario de energía, La demanda de energía ha incrementado un 3.5% en promedio en los últimos 30 años, por su parte la población se ha expandido un 1,2% y el PIB del mundo un 6% anual (BBVA Research, Sector eléctrico colombiano), lo que se traduce en facturas más elevadas para los usuarios. Esta situación no solo repercute en los costos económicos, sino que también contribuye significativamente al aumento de la demanda en la generación de energía eléctrica y, por ende, a mayores emisiones de dióxido de carbono (CO2), intensificando el impacto ambiental [1].

1.1.1. Planteamiento del problema

En la actualidad, el consumo energético se ha convertido en un factor crítico tanto a nivel ambiental como económico. Muchas edificaciones comerciales utilizan energía de manera ineficiente, lo que genera un aumento innecesario en el gasto energético y contribuye al deterioro del medio ambiente debido a la elevada emisión de gases de efecto invernadero. A pesar de la existencia de tecnologías y prácticas que permiten un uso más racional de la energía, su adopción es todavía limitada por falta de información, inversión inicial o conciencia sobre los beneficios a largo plazo [2].
Una de las principales barreras para mejorar la eficiencia energética es la falta de información detallada sobre cómo, cuándo y cuánto consumen los diferentes equipos eléctricos conectados a una red. Los sistemas de medición convencionales brindan datos generales del consumo total, pero no permiten identificar patrones de uso específicos ni detectar cargas ineficientes o mal funcionamiento de equipos eléctricos. Por otro lado, los sistemas de monitoreo intrusivos requieren sensores físicos en cada dispositivo, lo cual resulta costoso, complejo de instalar y poco escalable, llevando a los usuarios a no poder tener un control o análisis detallado de cual o cuales dispositivos pueden estar incrementando dicho consumo. Sin embargo, los datos recolectados por estos sistemas intrusivos pueden ser los insumos necesarios para realizar iteraciones entre modelos de machine learning que nos permitan desagregar el consumo eléctrico. En este contexto, resulta fundamental establecer un proceso de reconciliación entre las estimaciones generadas por los modelos y los datos reales agregados, con el fin de validar la consistencia de los resultados, ajustar los modelos y garantizar una representación fiel del comportamiento energético. Los modelos de deep learning en particular las redes neuronales recurrentes, nos permiten modelar series temporales y entender dichos patrones de consumo [3].

El desarrollo de un sistema de análisis de carga no intrusivo basado en modelos de machine learning con redes neuronales recurrentes, contribuiría significativamente a mejorar la eficiencia energética en diversos entornos, reducir los costos operativos y minimizar el impacto ambiental, al mismo tiempo que empodera a los usuarios con información útil para la toma de decisiones.
A través del aprendizaje automático, es posible descomponer el consumo eléctrico total registrado en un único punto de medición en estimaciones por tipo de carga, sin necesidad de instalar medidores individuales en cada equipo eléctrico. Esto se logra gracias a la capacidad de los modelos de machine learning para identificar patrones característicos de consumo asociados a distintas categorías de

dispositivos. Como resultado, se mejora la visibilidad del consumo energético desagregado, permitiendo decisiones más precisas y fundamentadas en datos reales.

El análisis conjunto de datos agregados y desagregados posibilita la detección de picos de consumo inusuales, uso de equipos eléctricos fuera del horario laboral y la identificación de categorías con consumos innecesarios o ineficientes. Esto contribuye directamente a la optimización del uso de energía, permitiendo intervenir en los puntos críticos del sistema eléctrico de manera focalizada.

Adicionalmente, los modelos predictivos permiten anticipar el comportamiento energético futuro de las oficinas, facilitando la planificación de la demanda y el diseño de estrategias de eficiencia adaptadas a las condiciones operativas específicas de cada entorno.

En conjunto, el uso de IA y ciencia de datos en este contexto no solo permite monitorear el consumo energético de manera más inteligente, sino que transforma datos en conocimiento útil para promover una cultura organizacional orientada a la sostenibilidad y la eficiencia operativa.

1.1.2. Formulación del problema

¿Cómo podemos analizar la eficiencia energética desagregando el consumo a través de un modelo de machine learning, sin recurrir a sistemas de medición intrusivos?

- ¿Qué tipos de datos permiten identificar patrones para la desagregación de equipos eléctricos de consumo eléctrico por dispositivo?
- ¿Qué tipo de algoritmos de machine learning resultan más efectivos para reconocer patrones de consumo eléctrico, y cómo puede garantizarse su precisión en entornos reales con múltiples cargas conectadas?
- ¿Cómo podemos generar un proceso de reconciliación que permita mejorar la coherencia y precisión entre los modelos de desagregación por categorías?
- ¿Qué métricas permiten evaluar la precisión del modelo en la disgregación de consumo por dispositivo?
- ¿Es posible a partir de los resultados del modelo contribuir a la toma de decisiones para la eficiencia energética?

1.1.3. Sistematización

1.1.3.1. Procesamiento y recolección de datos

El procesamiento y la recolección de datos de consumo eléctrico se llevaron a cabo mediante un análisis exploratorio y la identificación de correlaciones significativas entre variables. En primer lugar, se realizó la recolección de datos provenientes de fuentes confiables, como medidores inteligentes, que registran información detallada sobre el consumo eléctrico. Posteriormente, se efectuó una limpieza de los datos recolectados, eliminando valores atípicos, inconsistencias y registros faltantes con el fin de garantizar la calidad y fiabilidad de la información. A continuación, se desarrolló un análisis exploratorio de datos (EDA) para identificar patrones, tendencias y distribuciones que permitieron comprender el comportamiento del consumo energético. Finalmente, se calcularon correlaciones entre variables relevantes, con el propósito

de determinar relaciones significativas que sirvieron como base para la modelación y análisis posterior del consumo eléctrico.

1.1.3.2. Evaluar diferentes modelos de machine learning

La evaluación de diferentes modelos de machine learning para la desagregación del consumo energético se desarrolló en varias etapas orientadas a identificar el enfoque más efectivo. En primer lugar, se realizó la selección de técnicas de aprendizaje automático adecuadas, considerando tanto algoritmos supervisados como no supervisados que permitieron abordar la complejidad del consumo eléctrico. A continuación, se llevó a cabo la implementación inicial de modelos base, entre ellos regresión lineal, árboles de decisión y redes neuronales, con el fin de evaluar su desempeño preliminar frente a los datos disponibles. Posteriormente, se efectuó una comparación de los modelos mediante el uso de métricas estándar como precisión, recall, F1-score y error cuadrático medio (MSE), lo que permitió determinar su eficacia y capacidad predictiva. Con base en los resultados obtenidos, se procedió al ajuste de los modelos seleccionados, modificando los parámetros de entrada para optimizar su rendimiento. Finalmente, se realizó el entrenamiento definitivo de los modelos con los datos previamente procesados, incorporando técnicas de regularización para evitar el sobreajuste y posibilitar la clasificación de al menos cuatro categorías de equipos eléctricos susceptibles de desagregación energética.

1.1.3.3. Desarrollar un mecanismo de reconciliación.

El desarrollo de un mecanismo de reconciliación tuvo como propósito alinear las predicciones de consumo generadas por distintos modelos de machine learning, garantizando coherencia y precisión en el proceso de desagregación energética. Para ello, se diseñó e implementó un modelo avanzado, fundamentado en técnicas como redes neuronales profundas (deep learning) o modelos híbridos, que permitieron integrar y optimizar los resultados obtenidos por diversos algoritmos. En esta etapa, se llevó a cabo la clasificación de las cargas eléctricas según categorías previamente definidas, tales como equipos de cómputo, iluminación, sistemas de climatización y centros de datos, con el fin de establecer una estructura de referencia coherente para la desagregación. Finalmente, se efectuó la validación del modelo implementado utilizando un conjunto de datos etiquetados, lo que permitió evaluar su capacidad de clasificación y verificar la consistencia de las predicciones generadas para cada categoría de carga energética.

1.1.3.4. Evaluar la precisión y eficiencia del modelo

El modelo final se aplicó a distintos conjuntos de datos que representaban diversos tipos de cargas eléctricas, con el fin de evaluar su generalización y robustez frente a variaciones en patrones de consumo. Para cuantificar su desempeño se midió tanto la precisión del modelo, eficiencia operativa, evaluando el tiempo de procesamiento y la escalabilidad en distintos escenarios de carga y volumen de datos. Estos indicadores permitieron determinar la viabilidad del modelo para su implementación en entornos reales y guiaron ajustes adicionales para mejorar tanto su precisión como su rendimiento computacional.

1.1.3.5. Apropiación social del conocimiento y soporte a la toma de decisiones

A partir de los resultados de desagregación y reconciliación obtenidos por el modelo, se evaluó en qué medida la información generada puede traducirse en insumos prácticos para la toma de decisiones orientadas a la eficiencia energética. Para ello, se analizó la utilidad de los reportes desagregados por

categoría de carga como base para la formulación de recomendaciones de ahorro energético, la identificación de oportunidades de mantenimiento preventivo en equipos con consumos anómalos y el diseño de estrategias de gestión de la demanda dirigidas a usuarios finales, administradores de edificaciones y empresas distribuidoras de energía. De esta manera, se buscó verificar que el modelo no solo alcance un desempeño predictivo adecuado, sino que también genere conocimiento aplicable y socialmente apropiable por los distintos actores involucrados en la gestión energética del edificio.

1.2. Objetivos del proyecto

1.2.1. Objetivo General

Desarrollar un modelo de análisis de carga no intrusivo basado en técnicas de machine learning que permita desagregar el consumo energético de los equipos.

1.2.2. Objetivos Específicos

1. Procesar los datos de carga energética y consumo eléctrico a través del análisis exploratorio y correlaciones.
2. Evaluar diferentes modelos de machine learning para la desagregación e implementación de un modelo capaz de identificar y categorizar las cargas de los equipos eléctricos.
3. Desarrollar un mecanismo de reconciliación para alinear las predicciones de consumo generadas por distintos modelos según las categorías de carga, para coherencia y precisión en la desagregación.
4. Evaluar la precisión y eficiencia del modelo considerando diferentes tipos de cargas eléctricas.
5. Evaluar la contribución de los resultados del modelo a la toma de decisiones orientadas a la eficiencia energética, mediante la apropiación social del conocimiento generado.

1.3. Marco de referencia

1.3.1. Marco teórico

La eficiencia energética es un aspecto clave en el desarrollo sostenible, ya que permite reducir el consumo de energía sin disminuir la calidad de los servicios, contribuyendo así a la mitigación del cambio climático y a la optimización de recursos económicos. Uno de los principales retos para lograrla es la falta de visibilidad sobre el consumo individual de los dispositivos eléctricos, especialmente en entornos donde solo se cuenta con una medición total de energía.

Los sistemas de monitoreo de carga no intrusivo (NILM, por sus siglas en inglés) han surgido como una alternativa tecnológica viable para abordar esta problemática. Estos sistemas permiten descomponer la señal de consumo total registrada por un único medidor eléctrico, e identificar de forma indirecta las cargas conectadas, mediante técnicas avanzadas de procesamiento de señales y modelos de aprendizaje automático. A diferencia de los sistemas intrusivos, no requieren sensores individuales en cada aparato, lo que reduce significativamente los costos y la complejidad de instalación.

En este contexto, la inteligencia artificial (IA) juega un papel fundamental. Algoritmos como redes neuronales artificiales, máquinas de vectores de soporte (SVM), árboles de decisión, k-NN, entre otros, han demostrado ser efectivos para clasificar patrones eléctricos y reconocer firmas de dispositivos a partir de datos agregados. Además, técnicas de aprendizaje profundo, como las redes neuronales convolucionales (CNN) y recurrentes (RNN), están siendo ampliamente utilizadas para tareas de desagregación de energía debido a su capacidad para capturar patrones complejos en series temporales.

Por lo tanto, integrar la IA en sistemas NILM representa una estrategia innovadora y eficiente para proporcionar a los usuarios información detallada sobre su consumo energético, detectar usos ineficientes, e incentivar decisiones más informadas orientadas al ahorro y sostenibilidad.

1.3.1.1. Eficiencia energética y su importancia

La eficiencia energética se refiere a la capacidad de obtener los mismos resultados o servicios utilizando menos energía. Este concepto ha cobrado relevancia debido al crecimiento de la demanda energética, la dependencia de fuentes fósiles, y la necesidad de mitigar el cambio climático. Según el informe de la Agencia Internacional de Energía (IEA), mejorar la eficiencia energética podría reducir las emisiones de gases de efecto invernadero en un 40% para el año 2040 [4].

En sectores comerciales, una parte significativa del consumo energético ocurre sin control ni análisis específico, lo que genera sobrecostos, ineficiencias y desperdicio energético. Además, el consumidor promedio carece de herramientas efectivas para entender el desglose de su consumo eléctrico, lo que impide acciones informadas para ahorrar energía. En este contexto, surge la necesidad de herramientas tecnológicas que permitan una gestión inteligente del consumo, como los sistemas de monitoreo energético avanzados.

1.3.1.2. Monitoreo de energía: de lo intrusivo a lo inteligente

El monitoreo del consumo eléctrico puede realizarse mediante dos enfoques:

- Intrusivo (ILM): Requiere sensores en cada dispositivo eléctrico, lo que eleva costos y complica la instalación.
- No intrusivo (NILM): Se basa en el análisis de la señal eléctrica total del inmueble, sin sensores individuales.

El concepto de NILM fue propuesto por Hart en 1992, quien introdujo un sistema que descompone la carga total en consumos individuales a través del análisis de patrones eléctricos [3]. Esta técnica permite identificar la operación de distintos equipos eléctricos mediante sus firmas eléctricas características, como cambios de potencia, picos de arranque, ciclos de funcionamiento y variaciones armónicas.

NILM se ha convertido en una opción viable para proyectos a gran escala y para aplicaciones en edificios inteligentes y sistemas de gestión de energía industrial.

1.3.1.3. Componentes técnicos del NILM

Un sistema NILM completo comprende una cadena de procesamiento que va desde la adquisición de la señal eléctrica hasta la entrega del reporte desagregado. El primer componente es la recoleccion de datos, que consiste en la medición del consumo eléctrico total en el punto de entrada del edificio o instalación. La resolución temporal de esta medición determina el tipo de características que es posible extraer: sistemas de alta frecuencia (kHz) permiten detectar transitorios de encendido/apagado, mientras que sistemas de baja frecuencia (minutos u horas) capturan patrones de uso agregados. Este trabajo opera sobre datos horarios, lo que sitúa el análisis en el rango de resolución temporal baja, apropiado para la gestión energética a nivel de categorías de carga.

El segundo componente es el preprocesamiento de la señal, que incluye la detección y corrección de valores atípicos, la imputación de datos faltantes y la normalización de variables. En el contexto de este trabajo, el análisis exploratorio presentado en el Capítulo 2 identificó valores inconsistentes y gaps temporales en el dataset original (véase Tabla 4), los cuales fueron tratados mediante interpolación lineal. Este paso es crítico porque la calidad del preprocesamiento determina directamente la capacidad de los modelos para aprender patrones representativos del consumo real.

El tercer componente es el algoritmo de desagregación, que infiere el consumo individual de cada carga a partir de la señal total. En la literatura se distinguen dos enfoques principales: los métodos basados en eventos, que detectan transitorios de cambio de estado para identificar el encendido o apagado de un dispositivo, y los métodos basados en optimización o regresión, que modelan directamente la relación entre el consumo total y los consumos parciales como un problema de aprendizaje supervisado. Este trabajo se ubica en la segunda categoría, empleando modelos de regresión entrenados con datos etiquetados a nivel de categoría de carga.

El cuarto componente es la evaluación del desempeño, que cuantifica la precisión de las estimaciones del

sistema. Las métricas estándar en problemas de regresión (MAE, RMSE, R^2) son directamente aplicables y permiten comparar el desempeño entre modelos y variables. Hart (1992), en la formulación original del problema NILM, señaló que la exactitud del sistema debe evaluarse tanto a nivel de carga individual como de consumo total, criterio que este trabajo satisface mediante el análisis por variable y el proceso de reconciliación descrito en el Capítulo 4.

1.3.1.4. Fundamentos de la regresión línea múltiple

La regresión lineal múltiple es el método de aprendizaje supervisado más simple para modelar la relación entre una variable dependiente y un conjunto de variables predictoras (Hastie et al., 2009). El modelo asume que la variable objetivo Y puede expresarse como una combinación lineal de las variables predictoras X_1, X_2, ..., X_p más un término de error ε:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \varepsilon \quad (1)$$

donde los coeficientes β se estiman mediante el método de mínimos cuadrados ordinarios (OLS), minimizando la suma de cuadrados de los residuos. La solución matricial es $\hat{\beta} = (X^T X)^{-1} X^T y$. La validez de la solución OLS descansa en los supuestos de Gauss-Markov: linealidad de la relación, homocedasticidad de los residuos, independencia de las observaciones y ausencia de multicolinealidad perfecta entre las variables predictoras. Cuando alguno de estos supuestos se viola — como ocurre cuando la relación entre el consumo total y las cargas individuales tiene naturaleza no lineal — el desempeño del modelo se deteriora significativamente.

1.3.1.5. Máquinas de vectores de soporte para regresión (SVR)

Las Máquinas de Vectores de Soporte (SVM) fueron introducidas por Cortes y Vapnik (1995) como clasificadores de margen máximo. Su extensión al problema de regresión, denominada Support Vector Regression (SVR), fue desarrollada por Drucker et al. (1997). En SVR, el objetivo es encontrar una función $f(x) = w^T\phi(x) + b$ que se aproxime a los valores objetivo con un error no mayor a ε, al mismo tiempo que se maximiza el margen en el espacio de características.

El elemento central de SVR es la función de pérdida ε-insensible, que ignora errores menores que un umbral ε (la denominada ε-tube) y penaliza linealmente los errores que superan dicho umbral. Formalmente, el problema de optimización primal es:

$$\begin{aligned} &\text{Minimizar: } \tfrac{1}{2}||w||^2 + C\cdot\Sigma(\xi_i + \xi_i^*) \quad (2)\\ &\text{sujeto a: } y_i - w^T\phi(x_i) - b \le \varepsilon + \xi_i \\ &\qquad w^T\phi(x_i) + b - y_i \le \varepsilon + \xi_i^*, \quad \xi_i, \xi_i^* \ge 0 \end{aligned}$$

donde C es el parámetro de regularización que controla el equilibrio entre la penalización por errores y la

complejidad del modelo (norma de w), y ξ_i, ξ_i^* son variables de holgura que miden la desviación de los puntos fuera de la ε-tube.

La capacidad de SVR para manejar relaciones no lineales se obtiene mediante kernel: en lugar de trabajar directamente con las variables de entrada x, se mapea el espacio original a un espacio de mayor dimensión mediante una función ϕ(x), donde el producto escalar $\phi(x)^T\phi(x')$ puede evaluarse eficientemente como K(x, x') sin necesidad de calcular ϕ explícitamente. El kernel de base radial (RBF), empleado en este trabajo, se define como:

$$K(x, x_i) = \exp(-\gamma||x - x_i||^2) \quad (3)$$

donde γ > 0 controla el radio de influencia de cada punto de soporte: valores elevados de γ producen modelos más complejos con regiones de decisión más locales, mientras que valores bajos generalizan de forma más suave. La elección del kernel RBF es adecuada para el problema de desagregación energética porque no impone supuestos sobre la forma funcional de la relación entre el consumo total y los consumos parciales, permitiendo capturar patrones no lineales y no monótonos presentes en los datos horarios.

Los tres hiperparámetros determinantes del modelo SVR son: C (penalización por violación del margen), ε (ancho de la zona de tolerancia) y γ (escala del kernel RBF).

1.3.1.6. Bosques aleatorios (Random Forest)

Los bosques aleatorios (Random Forest) fueron formalizados por Breiman (2001) como un método de ensamble que combina un gran número de árboles de decisión entrenados de forma independiente. La idea central es que, aunque un árbol individual es altamente susceptible al sobreajuste (alta varianza), el promedio de muchos árboles de correlacionados produce un estimador con varianza sustancialmente menor y sesgo comparable.

El fundamento estadístico del método radica en el teorema de reducción de varianza por promediado: si se dispone de B variables aleatorias idénticamente distribuidas con varianza σ^2 y correlación por pares ρ, su media tiene varianza:

$$\mathrm{Var}(\bar{Y}) = \rho\sigma^2 + (1-\rho)\sigma^2/B \quad (4)$$

Al aumentar B, el segundo término tiende a cero, pero el primer término $\rho\sigma^2$ persiste. La clave de Random Forest es reducir ρ, la correlación entre árboles, mediante dos mecanismos complementarios:

(1) bootstrap aggregating (bagging), donde cada árbol se entrena sobre una muestra con reemplazamiento del conjunto de datos original, y (2) selección aleatoria de características, donde en cada nodo de división solo se evalúan m características elegidas al azar de entre las p disponibles (típicamente $m \approx \sqrt{p}$ para clasificación y $m \approx p/3$ para regresión). Esta segunda innovación es la contribución específica de Breiman (2001) respecto al bagging convencional, ya que impide que todos los árboles encuentren las mismas divisiones dominantes, reduciendo drásticamente ρ.
Una propiedad operacional valiosa del método es la estimación out-of-bag (OOB): como cada árbol se

entrena sobre aproximadamente el 63.2% de las muestras, el 36.8% restante puede usarse como conjunto de validación interno, proporcionando una estimación del error de generalización sin necesidad de un conjunto de validación separado. Esto resulta especialmente útil durante la fase de búsqueda de hiperparámetros.

1.3.1.7. Gradient boosting y XGBoost

El gradient boosting fue formalizado por Friedman (2001) como un método de ensamble secuencial en el que cada nuevo modelo débil se entrena para aproximar el gradiente negativo de la función de pérdida con respecto a las predicciones actuales del ensemble. A diferencia del bagging (que construye modelos en paralelo), el boosting construye los modelos de forma iterativa, donde cada árbol corrige los errores residuales del ensemble acumulado hasta ese momento. El modelo final es una suma aditiva de la forma:

$$\hat{F}(x) = \Sigma_k f_k(x), \quad f_k \in F \tag{5}$$

donde F es el espacio de árboles de regresión y cada f_k se obtiene minimizando la función de pérdida L(y, $\hat{F}$) en la dirección del gradiente negativo en el espacio funcional.

XGBoost (Chen y Guestrin, 2016) introduce una regularización explícita en la función objetivo que distingue al método de implementaciones clásicas de GBDT:

$$L(\phi) = \Sigma_i l(y_i, \hat{y}_i) + \Sigma_k \Omega(f_k), \quad \text{donde} \quad \Omega(f) = \gamma T + \tfrac{1}{2}\lambda||\omega||^2 \tag{6}$$

donde T es el número de hojas del árbol k, ω es el vector de pesos de las hojas, γ establece la ganancia mínima de pérdida para realizar una división (lo que equivale a una poda), y λ es el término de regularización L2 sobre los pesos de las hojas. Esta regularización penaliza la complejidad del modelo y reduce la tendencia al sobreajuste que es característica de los métodos de boosting con tasas de aprendizaje elevadas.

Adicionalmente, XGBoost incorpora técnicas de submuestreo de columnas (colsample_bytree) y filas (subsample) por árbol, similares al muestreo aleatorio de Random Forest pero aplicadas en el contexto secuencial del boosting. El parámetro learning_rate (también denominado shrinkage) escala la contribución de cada árbol al ensemble acumulado, actuando como un regulador adicional de la velocidad de aprendizaje.

La comparación entre Random Forest y XGBoost en este trabajo ilustra la distinción fundamental entre bagging y boosting: mientras Random Forest exhibe un Gap R más contenido (3.0 a 7.7 para las variables con mejor desempeño), XGBoost muestra valores de Gap R considerablemente más altos en algunas variables (hasta 37.5 en Other electricity), lo que evidencia una mayor propensión al sobreajuste en el contexto específico del dataset empleado. Este comportamiento es consistente con la literatura: el boosting puede memorizar el conjunto de entrenamiento cuando el espacio de hiperparámetros no está suficientemente restringido.

1.3.1.8. Redes neuronales recurrentes

Las redes neuronales recurrentes (RNN) están diseñadas para procesar secuencias de datos manteniendo un estado oculto h_t que se actualiza en cada paso temporal. La dinámica de la RNN estándar es $h_t = \tanh(W_h h_{t-1} + W_x x_t + b)$, donde x_t es la entrada en el instante t y h_{t-1} es el estado oculto previo. Sin embargo, las RNN estándar sufren el problema del gradiente desvaneciente: durante el entrenamiento por retro propagación a través del tiempo (BPTT), el gradiente del error se multiplica repetidamente por W_h, lo que provoca que decrezca exponencialmente para dependencias temporales largas (Hochreiter y Schmidhuber, 1997), imposibilitando el aprendizaje de patrones distantes en el tiempo.
La arquitectura LSTM (Long Short-Term Memory), introducida por Hochreiter y Schmidhuber (1997), resuelve este problema mediante un mecanismo de memoria explícita compuesto por tres compuertas y un estado de celda c_t que fluye a lo largo de la secuencia con modificaciones controladas. Las ecuaciones de actualización son:

Compuerta de olvido: $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
Compuerta de entrada: $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
Estado candidato: $\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$
Estado de celda: $c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$
Compuerta de salida: $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
Estado oculto: $h_t = o_t \odot \tanh(c_t)$

donde σ denota la función sigmoide, $\odot$ el producto elemento a elemento (Hadamard) y W^*, b^* son los pesos y sesgos aprendibles. La compuerta de olvido f_t determina qué información del estado de celda previo se retiene; la compuerta de entrada i_t controla qué información nueva se almacena; la compuerta de salida o_t filtra qué parte del estado de celda se expone como estado oculto. Este diseño permite que el gradiente fluya sin atenuar a través del estado de celda, habilitando el aprendizaje de dependencias temporales de largo alcance.

En el contexto de este trabajo, el modelo RNN-LSTM recibe como entrada una ventana deslizante de window_size pasos temporales del consumo total y otras variables derivadas, y predice el consumo de la variable objetivo en el instante siguiente. La hipótesis implícita es que el consumo en el instante t depende de los consumos previos en una ventana temporal relevante. Sin embargo, los resultados del trabajo evidencian que el RNN-LSTM no logra superar a los modelos de ensamble (R^2 Test máximo de 51% para Sockets plug, 76% para Ventilation), con valores de Gap R extremadamente elevados que indican sobreajuste severo. Esta limitación puede explicarse por las características del dataset: la variación del consumo horario en un edificio comercial está determinada principalmente por patrones determinísticos asociados al calendario (hora, día, mes) y no por dependencias temporales secuenciales de largo alcance. En este escenario, la capacidad de las LSTM para modelar secuencias largas no aporta ventaja sobre los modelos que capturan directamente las variables calendario como features, como hacen Random Forest y XGBoost (Goodfellow et al., 2016).

1.3.1.9. Técnicas de inteligencia artificial aplicadas al NILM

Los métodos de inteligencia artificial para NILM pueden clasificarse en tres grandes familias según su paradigma de aprendizaje y el tipo de representación del comportamiento de las cargas. La primera familia corresponde a los modelos probabilísticos, entre los que destacan los modelos ocultos de Markov (HMM) y su variante factorial (FHMM). Estos modelos representan el estado de cada electrodoméstico (encendido/apagado) como una variable latente y el consumo observado como una emisión probabilística

de la combinación de estados activos. Su principal ventaja es que no requieren datos etiquetados a nivel de dispositivo, ya que el aprendizaje se realiza en forma no supervisada o semi-supervisada. Sin embargo, su complejidad crece exponencialmente con el número de cargas modeladas (Kolter y Jaakkola, 2012; Bonfigli et al., 2017), lo que limita su aplicabilidad en edificios con muchas categorías de consumo.

La segunda familia comprende los modelos de aprendizaje profundo, que han dominado la literatura de NILM en los últimos años. Las redes neuronales convolucionales (CNN) extraen patrones locales de la señal de potencia, mientras que las redes neuronales recurrentes (RNN) y sus variantes LSTM y GRU capturan dependencias temporales secuenciales. Arquitecturas más recientes, como las basadas en mecanismos de atención tipo Transformer (Azad et al., 2023), han demostrado capacidad para modelar dependencias de largo alcance en señales de alta frecuencia. El principal requisito de estos métodos es disponer de grandes volúmenes de datos etiquetados y señales de alta resolución temporal para que la representación temporal aprendida sea significativa (Kelly y Knottenbelt, 2015; Zhang et al., 2017).

La tercera familia, en la que se enmarca este trabajo, corresponde a los modelos de regresión supervisada, particularmente los métodos de ensamble como Random Forest y XGBoost. Estos modelos son apropiados cuando se dispone de datos etiquetados a nivel de categoría de carga y la resolución temporal es horaria o diaria, condiciones que eliminan la ventaja de los modelos secuenciales profundos al no existir dependencias temporales de corta escala. En el contexto de edificios comerciales con medición horaria, la variación del consumo está principalmente explicada por variables determinísticas como la hora del día, el día de la semana y la época del año, que los modelos de ensamble capturan directamente como características sin necesidad de arquitecturas recurrentes. Xiao et al. (2021) confirmaron esta idoneidad al obtener $R^2 = 0.88$ para desagregación de cooling en edificios inteligentes empleando Random Forest.

1.3.1.10. Retos del monitoreo no intrusivo

A pesar de los avances, los sistemas NILM enfrentan diversos desafíos:

- Ambigüedad entre dispositivos: Aparatos similares (por ejemplo, varios ventiladores) pueden tener firmas eléctricas casi idénticas.
- Ruidos en la red eléctrica: Variaciones aleatorias y ruido dificultan la detección precisa.
- Cambios en el entorno: Nuevas cargas o dispositivos modifican el patrón total, afectando la precisión del sistema.
- Falta de datos etiquetados: Los modelos supervisados requieren ejemplos reales, lo cual es costoso de obtener.

Soluciones emergentes incluyen el transfer learning, técnicas semi-supervisadas, y entrenamiento con datos sintéticos generados por IA, que permiten adaptar modelos a nuevos contextos sin volver a etiquetar manualmente los datos.

1.3.1.11. Aplicaciones prácticas y beneficios

La implementación de un sistema NILM con inteligencia artificial permite:

- Optimizar el consumo energético en edificaciones comerciales.

- Detectar equipos eléctricos defectuosos o ineficientes.
- Promover hábitos de consumo consciente en los usuarios.
- Facilitar esquemas de respuesta a la demanda (Demand Response).
- Apoyar políticas públicas de sostenibilidad energética.
- Desarrollar sistemas de domótica o automatización adaptativos.

Además, es posible integrar estas tecnologías en plataformas IoT, dashboards

1.3.1.12. Métricas de evaluación del desempeño

Adicionalmente, se analizaron las métricas empleadas en dichos estudios para la evaluación y comparación del desempeño de los modelos. A partir de esta información, se seleccionaron como métricas de evaluación el error cuadrático medio (RMSE), el error absoluto medio (MAE) y el coeficiente de determinación (R^2), considerando tanto los resultados obtenidos en el conjunto de entrenamiento como en el conjunto de prueba, con el fin de evaluar la capacidad de ajuste y generalización de los modelos.

El RMSE mide la magnitud promedio de los errores entre los valores observados y los valores predichos por el modelo, otorgando mayor peso a los errores de mayor magnitud debido a la elevación al cuadrado de las diferencias. Se define como:

$$RMSE = \sqrt{\frac{1}{n}\sum(y_i - \hat{y}_i)^2} \quad (7)$$

donde y_i representa el valor observado, $\hat{y}_i$ el valor estimado por el modelo y n el número total de observaciones.

El MAE corresponde al promedio de las diferencias absolutas entre los valores observados y los valores predichos, proporcionando una medida directa del error promedio del modelo. Su expresión es:

$$MAE = \frac{1}{n}\sum|y_i - \hat{y}_i| \quad (8)$$

Finalmente, el coeficiente de determinación (R^2) cuantifica la proporción de la variabilidad de la variable dependiente que es explicada por el modelo. Valores cercanos a 1 indican un mejor ajuste del modelo a los datos observados. Se calcula como:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (9)$$

1.3.2. Reconciliación de pronósticos en series temporales jerárquicas

En muchas aplicaciones de pronóstico se trabaja con colecciones de series temporales que presentan una estructura jerárquica natural, es decir, donde algunas series son la suma de otras. En el contexto de la

desagregación energética, el consumo total de un edificio es la suma de los consumos individuales de sus categorías de carga. Cuando se construyen modelos independientes para cada nivel de la jerarquía, es frecuente que los pronósticos resultantes no sean coherentes: la suma de las predicciones de los subcomponentes no coincide con la predicción del total, violando una restricción física fundamental (Hyndman et al., 2011).

Formalmente, en una jerarquía de dos niveles — un agregado y n subcomponentes — la restricción de coherencia exige que:

$$y_t = \Sigma_{i=1}^{n} b_{it} \quad (10)$$

donde y_t es el valor del nivel agregado y b_{it} son los valores de los n subcomponentes en el instante t. Esta restricción puede expresarse matricialmente como $y_t = S \cdot b_t$, donde S es la matriz de suma (summing matrix) de dimensión (n+1)×n que describe la estructura de la jerarquía (Hyndman y Athanasopoulos, 2021).
La literatura propone tres familias de métodos de reconciliación. El método bottom-up agrega los pronósticos de los subcomponentes para obtener el pronóstico del total, lo que garantiza coherencia pero no aprovecha la información del nivel agregado durante el entrenamiento. El método top-down distribuye el pronóstico del total entre los subcomponentes según proporciones históricas, capturando información del nivel superior, pero dependiendo de la estabilidad de dichas proporciones. Los métodos de combinación óptima (Hyndman et al., 2011) minimizan el error cuadrático conjunto de todos los pronósticos reconciliados bajo la restricción de coherencia, expresándose como:

$$\tilde{y}_t = SG \cdot \hat{y}_t$$

donde $\tilde{y}_t$ es el vector de pronósticos reconciliados, $\hat{y}_t$ el vector de pronósticos base de todos los niveles, y G es una matriz de ponderación estimada a partir de los residuos históricos. Wickramasuriya et al. (2019) propusieron el método MinT (Minimum Trace), que estima G minimizando la traza de la matriz de varianza-covarianza de los errores de reconciliación, proporcionando la solución óptima insesgada.
En este trabajo se emplea un método de reconciliación proporcional, que constituye un caso especial del enfoque top-down: cada predicción individual y_i se ajusta multiplicándola por el cociente entre el consumo total real observado (y_{total}) y el total predicho ($\hat{y}_{total} = \Sigma y_i$), generando el valor reconciliado $yR_i = (y_i / \hat{y}_{total}) \times y_{total}$. Este procedimiento garantiza la coherencia energética por construcción ($\Sigma\, yR_i = y_{total}$) conservando las proporciones relativas entre las predicciones base. Si bien este método no minimiza el error de reconciliación en el sentido de MinT, su implementación es directa, computacionalmente trivial y no requiere estimar la estructura de covarianza de los errores, lo que lo hace adecuado como primera aproximación en contextos donde el objetivo principal es garantizar la consistencia con el principio físico de conservación de energía.

1.3.3. Antecedentes

El monitoreo no intrusivo de cargas (NILM, por sus siglas en inglés) ha sido objeto de creciente interés en los últimos años, como una estrategia efectiva para mejorar la eficiencia energética sin recurrir a instalaciones costosas o invasivas. Diversos estudios han explorado enfoques que combinan técnicas de procesamiento de señales eléctricas con algoritmos de inteligencia artificial, con el objetivo de identificar y clasificar el consumo de dispositivos individuales a partir de mediciones agregadas. En esta sección se presentan trabajos relevantes que han abordado el problema de la desagregación energética desde distintas perspectivas metodológicas, incluyendo redes neuronales profundas, modelos estadísticos, y revisiones sistemáticas del estado del arte. Cada estudio es analizado en términos de su contribución

específica, su enfoque técnico y sus diferencias con el proyecto que se propone, permitiendo identificar oportunidades para avanzar hacia una solución más práctica, escalable y centrada en el usuario.

1.4. Estado del arte

El campo del monitoreo de carga no intrusivo (NILM) ha experimentado una evolución significativa desde su formulación original por Hart (1992), transitando desde enfoques basados en umbrales de potencia hacia métodos estadísticos avanzados, modelos probabilísticos y arquitecturas de aprendizaje profundo. En esta sección se presenta una revisión sistemática de los desarrollos más relevantes, con énfasis en los trabajos que emplean técnicas de machine learning para la desagregación de carga en entornos no residenciales, los cuales constituyen el antecedente directo de este trabajo.

1.4.1 Métodos clásicos y probabilísticos

Los primeros enfoques de NILM modernos adoptaron modelos probabilísticos para representar el comportamiento de los electrodomésticos. Kolter y Jaakkola (2012) propusieron el uso de modelos ocultos de Markov factoriales (FHMM) para desagregación energética residencial, estableciendo una línea de trabajo que modelaba explícitamente los estados ON/OFF de cada carga. Aunque estos métodos ofrecen interpretabilidad, su escalabilidad se ve limitada por el crecimiento exponencial del espacio de estados con el número de electrodomésticos. Bonfigli et al. (2017) extendieron este enfoque incorporando potencia reactiva al modelo aditivo de Markov, mejorando la discriminación entre cargas con firmas espectrales similares.

1.4.2. Métodos basados en aprendizaje profundo

Kelly y Knottenbelt (2015) introdujeron el concepto de Neural NILM, demostrando que redes neuronales profundas podían superar a los métodos clásicos en precisión de desagregación sobre el dataset UK-DALE. Zhang et al. (2017) propusieron la arquitectura Sequence-to-Point (S2P), en la que una red neuronal recibe una ventana de señal agregada y predice el consumo puntual de un electrodoméstico específico. Este enfoque estableció una base metodológica ampliamente adoptada. Azad et al. (2023) incorporaron mecanismos de atención tipo Transformer, logrando mejoras en la captura de dependencias temporales largas, si bien a costa de una mayor exigencia computacional y de datos.

1.4.3. Aplicaciones en edificios comerciales con modelos de ensamble

El trabajo de Xiao et al. (2021) es especialmente relevante para esta investigación, ya que aplica Random Forest para desagregación de carga de enfriamiento (cooling) en edificios inteligentes, reportando un R^2 de 0.88 y demostrando la viabilidad de modelos de ensamble en contextos no residenciales. Zhou et al. (2021) propusieron un modelo híbrido CNN-LSTM-RF para NILM, obteniendo resultados competitivos en variables de consumo de equipos eléctricos y señalando la complementariedad entre características locales y temporales. Rajkumar et al. (2025) compararon directamente FHMM y LSTM para gestión energética inteligente, encontrando que LSTM supera a FHMM en precisión, pero requiere mayor esfuerzo de ajuste y datos de entrenamiento etiquetados.

- Es el artículo A **Survey on Non-Intrusive Load Monitoring Methods and Techniques for Energy Disaggregation Problem** se presenta una revisión detallada de las metodologías empleadas en NILM, desde técnicas estadísticas como los modelos ocultos de Markov

hasta algoritmos de machine learning supervisados y no supervisados. El trabajo proporciona una visión amplia del estado del arte, discutiendo las fortalezas, limitaciones y desafíos como la necesidad de datos etiquetados y la evaluación estandarizada. Si bien es una fuente teórica fundamental que permite entender el desarrollo histórico y técnico del NILM, no propone un sistema práctico ni soluciones aplicadas. En este sentido, el proyecto actual se apoya en este análisis para tomar decisiones informadas sobre qué modelos usar, pero avanza hacia la implementación real de un sistema de monitoreo energético basado en inteligencia artificial, aplicable a usuarios finales [10].

- El trabajo **Non-intrusive load monitoring by using active and reactive power in additive factorial hidden Markov models** propone un enfoque NILM utilizando tanto potencia activa como reactiva, modeladas mediante factorial hidden Markov models (FHMM), para mejorar la identificación de dispositivos eléctricos en ambientes donde múltiples cargas están activas simultáneamente. Se demuestra que incluir variables adicionales como la potencia reactiva mejora la precisión del modelo frente a métodos que consideran solo la potencia activa. Aunque se trata de una solución efectiva desde el punto de vista técnico, su aplicación práctica puede ser limitada por la necesidad de mayor complejidad en el monitoreo de señales y procesamiento de datos. A diferencia de esta propuesta centrada en modelos estadísticos, el presente proyecto opta por aprovechar redes neuronales profundas que requieren menos ingeniería manual de características, con el objetivo de construir un sistema adaptable y escalable que pueda ser desplegado en escenarios reales con menor costo de instalación [11].

- El estudio **Neural NILM: Deep Neural Networks Applied to Energy Disaggregation** es pionero aplica redes neuronales profundas, como CNNs y LSTMs, al problema de NILM, utilizando datasets públicos como REDD y UK-DALE. Se demuestra que estos modelos pueden superar a métodos tradicionales en precisión al modelar relaciones temporales y espaciales complejas dentro de las señales de consumo eléctrico. Los autores prueban diferentes arquitecturas y analizan su capacidad para desagregar la señal total en consumos individuales de equipos eléctricos. Aunque este trabajo representa un avance clave en la aplicación de deep learning, se limita a experimentos de laboratorio y no considera la adaptación del sistema a entornos con datos limitados, ruidos o variaciones no controladas. Nuestro proyecto, en cambio, busca llevar estos avances al terreno práctico, combinando modelos de IA con estrategias de adquisición de datos y despliegue en entornos reales, incluso con restricciones de hardware o calidad de datos [12].

2. PROCESAMIENTO Y RECOLECCIÓN DE DATOS DE CARGA ENERGÉTICA Y CONSUMO ELÉCTRICO A TRAVÉS DEL ANÁLISIS EXPLORATORIO Y CORRELACIONES.

2.1. Introducción

El Non-Intrusive Load Monitoring (NILM) es una técnica ampliamente utilizada para la desagregación del consumo energético en sistemas eléctricos, la cual permite estimar el consumo de diferentes cargas a partir de una única medición agregada registrada en el punto de entrada del sistema eléctrico. A diferencia de los métodos intrusivos, que requieren la instalación de sensores individuales en cada dispositivo o circuito eléctrico, el enfoque NILM se basa en el análisis de patrones presentes en la señal total de consumo eléctrico para identificar y estimar la contribución de diferentes categorías de carga. Debido a estas características, esta metodología ha adquirido una creciente relevancia en el ámbito de la eficiencia energética, la gestión inteligente de la energía y el análisis de patrones de consumo en edificaciones.

En el contexto de esta investigación, el enfoque NILM se emplea para identificar y pronosticar el consumo energético asociado a diferentes categorías de uso final de la energía dentro de un edificio comercial. En particular, se consideran las categorías de enfriamiento, calefacción, iluminación, enchufes, ventilación y otros usos eléctricos, las cuales representan componentes relevantes del consumo energético total en este tipo de instalaciones. La desagregación por categorías permite obtener una representación más detallada del comportamiento energético del edificio, facilitando la identificación de los sistemas con mayor demanda energética y proporcionando información útil para el análisis del consumo.

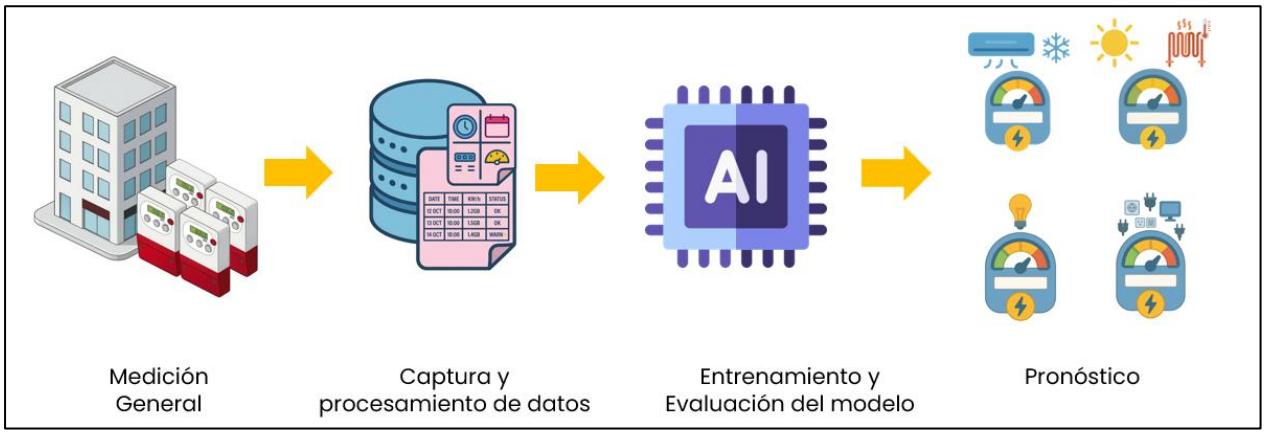


Imagen 1. Metodología de desagregación y pronóstico energético basada en NILM

En la Imagen 1. se observa la metodología de desagregación empleada en esta investigación, la cual se describe a continuación.

La primera etapa del proceso corresponde a la medición del consumo energético de cada categoría mediante el uso de medidores inteligentes, los cuales registran la potencia eléctrica asociada a cada

categoría en intervalos temporales de una hora. Esta información constituye la base de datos utilizada para el análisis posterior y permite disponer de series temporales representativas del comportamiento energético de las diferentes categorías consideradas en el estudio.

Posteriormente, se desarrolla la etapa de captura y procesamiento de los datos. Durante esta fase, los registros obtenidos del sistema de medición son almacenados y sometidos a diferentes procedimientos de preprocesamiento con el objetivo de mejorar la calidad de la información y facilitar su utilización en modelos de análisis. Entre las tareas realizadas se incluyen la eliminación de registros duplicados, el tratamiento de valores atípicos, la imputación de datos faltantes mediante técnicas de interpolación y la normalización de las variables. Asimismo, se realiza un proceso de regularización temporal para garantizar que la serie de datos presente intervalos de muestreo uniformes, lo cual resulta fundamental para el análisis de series temporales y el entrenamiento de modelos predictivos.

La tercera etapa corresponde al entrenamiento y evaluación del modelo de aprendizaje automático. En esta fase se aplican algoritmos de inteligencia artificial capaces de aprender las relaciones existentes entre las variables de entrada y los patrones de consumo energético asociados a cada categoría. Diversos enfoques han sido propuestos en la literatura para abordar este problema, incluyendo métodos de aprendizaje automático tradicionales y técnicas de aprendizaje profundo. El desempeño de los modelos desarrollados se evalúa mediante métricas estadísticas que permiten cuantificar la precisión de las estimaciones generadas y comparar el comportamiento de los diferentes enfoques utilizados.

Finalmente, se desarrolla la etapa de pronóstico o estimación del consumo energético por categoría. En esta fase, el modelo previamente entrenado se emplea para estimar el consumo individual de las distintas categorías eléctricas presentes en la instalación, tales como sistemas de enfriamiento, calefacción, iluminación, enchufes, ventilación y otros usos eléctricos. Los resultados obtenidos permiten identificar patrones de consumo energético, detectar posibles ineficiencias en el uso de la energía y generar información relevante para apoyar la toma de decisiones orientadas a mejorar la eficiencia energética del edificio.

En este contexto, los sistemas NILM se consolidan como una herramienta de gran utilidad para el análisis detallado del consumo eléctrico y la gestión eficiente de la energía en edificaciones inteligentes, permitiendo transformar datos agregados de consumo en información valiosa para la optimización energética.

El desarrollo de este capítulo responde al primer objetivo específico del proyecto, orientado al procesamiento y preparación de los datos como etapa fundamental para la construcción de modelos de desagregación energética. En problemas de aprendizaje automático, la calidad de los datos influye directamente sobre la capacidad predictiva de los modelos; por ello, antes de cualquier proceso de entrenamiento fue necesario realizar un análisis exploratorio exhaustivo que permitiera comprender la estructura del conjunto de datos, identificar relaciones entre variables, detectar valores atípicos, evaluar inconsistencias temporales y garantizar la consistencia de la información utilizada posteriormente. Las decisiones tomadas durante esta etapa constituyen la base metodológica que sustenta el desempeño obtenido por los modelos evaluados en los capítulos siguientes.

2.2. Análisis exploratorio y limpieza de datos

2.2.1. Análisis exploratorio (EDA)

El conjunto de datos utilizado en esta investigación proviene de una fuente pública disponible en la web y corresponde a las mediciones registradas por seis medidores inteligentes instalados en un edificio comercial. La información cubre un periodo continuo de cinco años, comprendido entre el 3 de enero de 2017 a las 13:00 horas y el 3 de enero de 2022 a las 12:00 horas, lo que permite capturar patrones de consumo energético a corto, mediano y largo plazo. La amplitud temporal del conjunto de datos resulta especialmente relevante para el análisis de carga no intrusiva (NILM), ya que posibilita la identificación de comportamientos operativos asociados a ciclos horarios, semanales y estacionales.

El conjunto de datos contiene 43824 registros con resolución horaria y presenta mediciones del consumo energético desagregado por tipo de carga eléctrica (véase Tabla 1). La variable Time indica la fecha y hora de cada observación. Las variables Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating representan el consumo de energía en kWh correspondiente a los sistemas de ventilación, equipos conectados a los enchufes, iluminación, otros consumos eléctricos no clasificados, refrigeración y calefacción, respectivamente. Esta desagregación permite analizar de manera individual el comportamiento de cada sistema y evaluar su contribución al consumo energético total del edificio.

La disponibilidad de mediciones separadas por tipo de carga constituye una ventaja para el desarrollo de modelos de análisis de carga no intrusiva, ya que facilita la caracterización de patrones de uso, la identificación de periodos de alta demanda y la comprensión de las relaciones existentes entre los diferentes sistemas consumidores de energía. Asimismo, la granularidad horaria de los datos proporciona un nivel de detalle adecuado para estudiar la dinámica temporal del consumo y establecer una base sólida para las etapas posteriores de limpieza, modelado y validación.

Tabla 1. Encabezado del conjunto de datos.

index	Time	Ventilation	Sockets plug	Lighting	Other electricity	Cooling	Heating
0	2017-01-03 13:00:00	27.4	41.8	53.7	89.6	18.6	498.0
1	2017-01-03 14:00:00	21.6	37.5	50.9	64.6	61.9	500.0
2	2017-01-03 15:00:00	18.5	37.4	60.1	64.2	16.5	480.0
3	2017-01-03 16:00:00	29.7	38.0	52.6	43.2	19.8	390.0
4	2017-01-03 17:00:00	16.7	38.1	56.7	42.4	15.2	373.0

Durante la exploración inicial de los datos no se identificaron valores faltantes, lo que indica una adecuada completitud de la información registrada por los medidores inteligentes durante el periodo de estudio. El análisis estadístico descriptivo (véase Tabla 2) evidencia diferencias importantes en el comportamiento de las cargas eléctricas monitoreadas. Las variables Other electricity y Heating presentan los mayores consumos promedio, con valores de 87.85 kWh y 68.99 kWh, respectivamente, lo que sugiere que estos sistemas constituyen una proporción significativa de la demanda energética total del edificio. Por el contrario, Lighting registra el menor consumo medio (14.32 kWh), reflejando una menor participación

relativa dentro del perfil energético global.

La dispersión de los datos, medida mediante la desviación estándar, revela una elevada variabilidad en las variables Heating (113.33 kWh), Other electricity (94.95 kWh) y Cooling (71.42 kWh). Este comportamiento indica que dichas cargas experimentan fluctuaciones importantes a lo largo del tiempo, posiblemente asociadas a cambios en las condiciones climáticas, variaciones en la ocupación del edificio o modificaciones en los requerimientos operativos. En contraste, variables como Lighting y Sockets plug presentan una menor dispersión, lo que sugiere patrones de consumo más estables y predecibles.

Los valores máximos observados en Cooling (1641 kWh) y Heating (1120 kWh) son considerablemente superiores a sus respectivos valores promedio y medianas, evidenciando la existencia de episodios de alta demanda energética. Estos picos de consumo pueden estar relacionados con periodos de condiciones climáticas extremas que incrementan el uso de los sistemas de climatización. Asimismo, la diferencia entre la media y la mediana en varias variables indica la presencia de distribuciones asimétricas, influenciadas por eventos de consumo excepcionalmente altos.

Por otra parte, todas las variables presentan valores mínimos iguales a cero, lo que confirma la existencia de intervalos temporales en los cuales determinados sistemas permanecieron inactivos. Este comportamiento es consistente con la naturaleza operativa de una infraestructura comercial, donde algunos equipos funcionan únicamente bajo condiciones específicas de ocupación o necesidad. En conjunto, los resultados muestran que el consumo energético del edificio presenta una combinación de cargas base relativamente constantes y cargas altamente variables, característica que deberá considerarse en las etapas posteriores de limpieza, análisis temporal y modelado del sistema de análisis de carga no intrusiva.

Tabla 2. Resumen del conjunto de datos.

index	Ventilation	Sockets plug	Lighting	Other electricity	Cooling	Heating
count	43824.0	43824.0	43824.0	43824.0	43824.0	43824.0
mean	32.43	18.41	14.32	87.85	47.08	68.99
std	36.21	27.67	15.06	94.95	71.42	113.33
min	0.0	0.0	0.0	0.0	0.0	0.0
25%	7.0	11.4	3.69	35.8	13.6	5.5
50%	14.1	13.6	6.5	79.6	21.5	12.3
75%	56.8	20.1	22.5	103.0	51.6	89.0
max	793.0	980.0	132.0	971.0	1641.0	1120.0

Con el fin de evaluar la calidad del conjunto de datos, se realizó un análisis de valores atípicos e inconsistencias temporales. Los valores atípicos fueron identificados mediante técnicas estadísticas de detección de anomalías aplicadas a cada variable de consumo energético. Como resultado, se encontraron 16958 registros atípicos, equivalentes al 38.7 % del total de observaciones (véase Tabla 3). Las variables Cooling (10.4 %) y Heating (11.25 %) concentraron la mayor proporción de estos registros, seguidas por

Sockets plug (8.07 %), mientras que Ventilation presentó la menor incidencia (1.02 %).

La elevada presencia de valores atípicos en los sistemas de climatización puede asociarse a incrementos extraordinarios de demanda durante periodos de condiciones climáticas extremas, donde los equipos de calefacción y refrigeración operan con mayor intensidad. Por esta razón, dichos registros no deben interpretarse automáticamente como errores de medición, ya que pueden representar eventos reales de consumo energético. No obstante, su magnitud y frecuencia justifican la aplicación de procedimientos de limpieza que permitan reducir posibles distorsiones en el análisis exploratorio y en las etapas posteriores de modelado.

Tabla 3. Resumen valores atípicos.

Variable	Cantidad Outliers	Porcentaje (%)
Ventilation	448	1.02
Sockets plug	3537	8.07
Lighting	1334	3.04
Other electricity	2154	4.92
Cooling	4556	10.4
Heating	4929	11.25
Total	16958	38.7

Adicionalmente, se verificó la consistencia cronológica de la serie temporal mediante el análisis de la secuencia de la variable Time. Los resultados evidenciaron la existencia de diez registros con inconsistencias temporales (véase Tabla 4), correspondientes a observaciones duplicadas o a desfases de dos horas respecto a la frecuencia horaria esperada. Estas anomalías se presentan principalmente en fechas asociadas a cambios de horario estacional, donde es habitual encontrar repeticiones o saltos en los registros temporales debido a ajustes oficiales de la hora.

Aunque la cantidad de inconsistencias detectadas representa una fracción mínima respecto al total de observaciones, su presencia puede afectar la continuidad de la serie temporal y generar errores en procedimientos posteriores de análisis y modelado. En consecuencia, fue necesario realizar un proceso de depuración y regularización temporal con el propósito de garantizar la coherencia cronológica del conjunto de datos y preservar la calidad de la información utilizada en el desarrollo de la investigación.

Tabla 4. Registros inconsistencias temporales.

Time	gap
2017-01-03 13:00:00	NaT
2017-03-26 03:00:00	0 days 02:00:00
2017-10-29 02:00:00	0 days 00:00:00
2018-03-25 03:00:00	0 days 02:00:00
2018-10-28 02:00:00	0 days 00:00:00
2019-03-31 03:00:00	0 days 02:00:00
2019-10-27 02:00:00	0 days 00:00:00
2020-03-29 03:00:00	0 days 02:00:00
2020-10-25 02:00:00	0 days 00:00:00
2021-03-28 03:00:00	0 days 02:00:00
2021-10-31 02:00:00	0 days 00:00:00

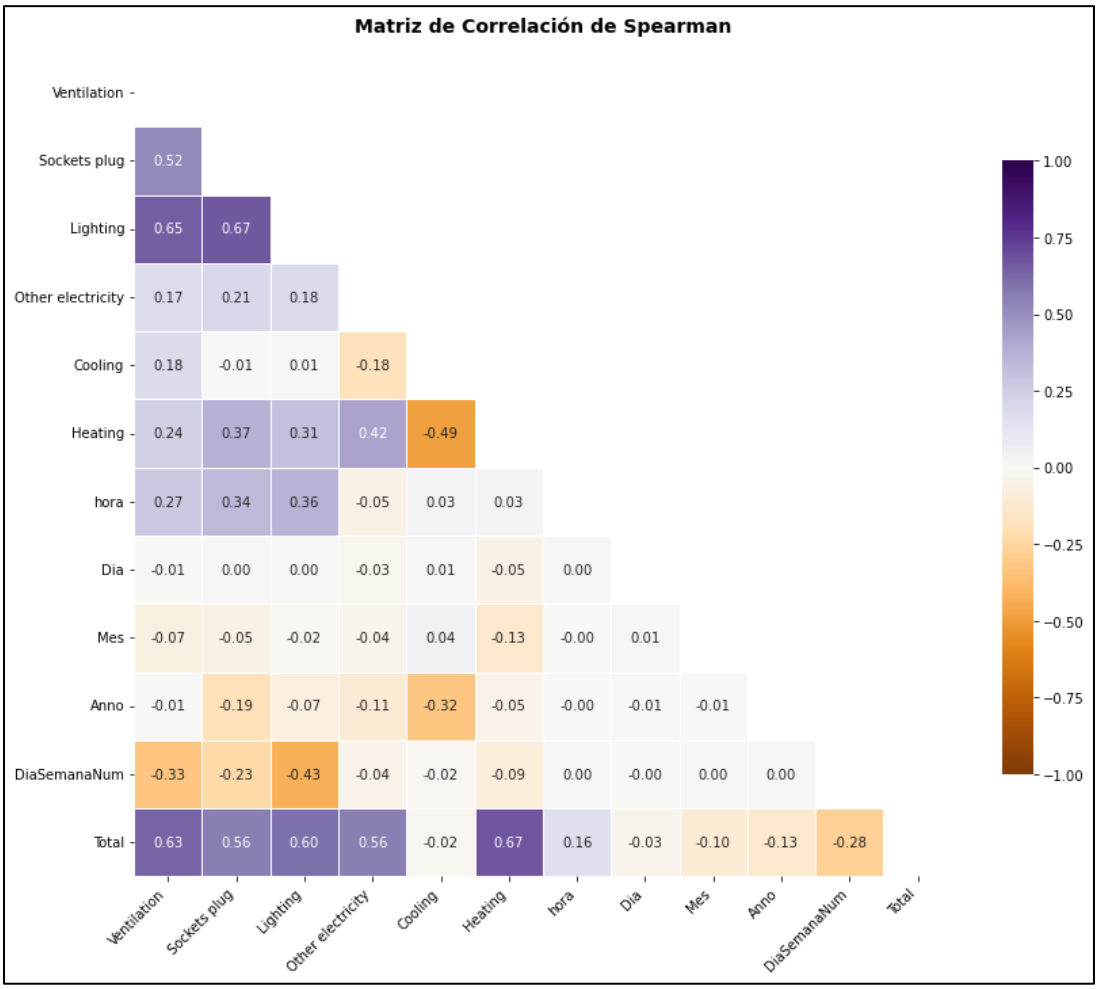


Imagen 2. Matriz de Correlación de Spearman.

La Imagen 2 presenta la matriz de correlación de Spearman calculada entre las variables de consumo energético y las variables temporales derivadas de la fecha y hora de registro. Este análisis permite identificar relaciones monotónicas entre las variables sin asumir distribuciones normales, siendo especialmente adecuado para datos energéticos que suelen presentar comportamientos no lineales y valores extremos.

En relación con las variables temporales, la variable Hora mostró correlaciones positivas con Ventilation (ρ=0.27), Sockets plug (ρ=0.34) y Lighting (ρ=0.36), indicando que el consumo de estas cargas tiende a incrementarse durante determinados momentos de la jornada. Este comportamiento es consistente con la operación de una infraestructura comercial, donde la ocupación y las actividades laborales se concentran principalmente en horarios diurnos. De manera complementaria, la variable DiaSemanaNum presentó una correlación moderada con Lighting (ρ=0.43), lo que evidencia diferencias significativas entre los días laborables y los fines de semana, confirmando la existencia de patrones operativos recurrentes asociados al uso del edificio.

Las correlaciones entre las variables de consumo energético permiten identificar grupos de cargas con comportamientos similares. Las relaciones más fuertes se observaron entre Lighting y Sockets plug (ρ=0.67), Lighting y Ventilation (ρ=0.65), así como Ventilation y Sockets plug (ρ=0.52). Estos resultados sugieren que dichos sistemas responden a factores comunes relacionados con la ocupación de la infraestructura, ya que la presencia de usuarios suele generar simultáneamente demanda de iluminación, ventilación y utilización de equipos eléctricos conectados a los enchufes.

Por otra parte, Heating presentó correlaciones positivas con Ventilation (ρ=0.24), Sockets plug (ρ=0.37), Lighting (ρ=0.31) y Other electricity (ρ=0.42), lo que indica que los periodos de mayor actividad operativa también coinciden con incrementos en los requerimientos de climatización. Adicionalmente, se observó una correlación negativa moderada entre Heating y Cooling (ρ=-0.49), comportamiento esperado debido a la naturaleza complementaria de estos sistemas, donde el aumento en la demanda de calefacción suele producirse en periodos de baja utilización de los sistemas de refrigeración y viceversa.

Al analizar la relación de cada variable con el consumo energético total, se encontró que Heating presenta la correlación más alta (ρ=0.67), seguida de Ventilation (ρ=0.63), Lighting (ρ=0.60), Sockets plug (ρ=0.56) y Other electricity (ρ=0.56). Estos resultados indican que dichas variables constituyen los principales contribuyentes a las variaciones de la demanda energética global del edificio. En contraste, Cooling mostró una correlación prácticamente nula con el consumo total (ρ=-0.02), sugiriendo un comportamiento más estacional y localizado en periodos específicos del año.

Desde la perspectiva del modelado, los resultados evidencian que el consumo energético no se comporta como un conjunto de variables independientes, sino como un sistema interrelacionado influenciado por factores temporales, operativos y climáticos. La identificación de estas dependencias permite comprender mejor la estructura del conjunto de datos y proporciona información valiosa para la selección de variables, la construcción de características temporales y el desarrollo de modelos de análisis de carga no intrusiva (NILM), donde las relaciones entre las diferentes cargas constituyen una fuente importante de información para la desagregación energética.

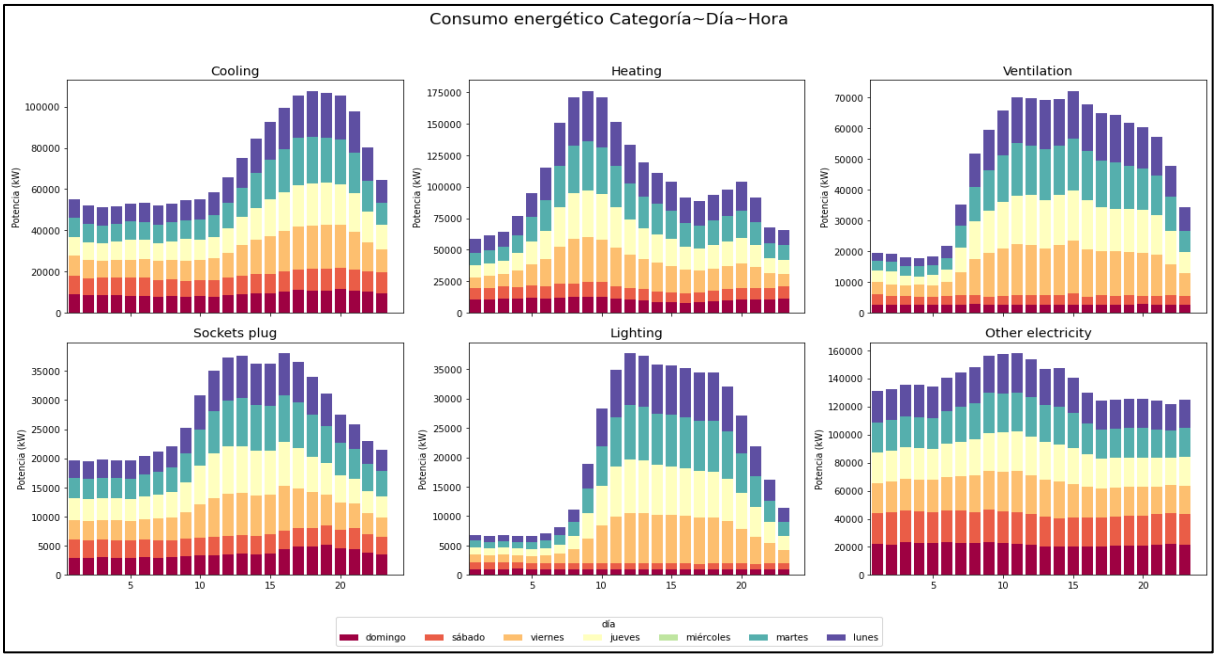


Imagen 3. Consumos por día-hora.

La Imagen 3 presenta la distribución del consumo energético en función de la hora del día y del día de la semana para cada una de las categorías analizadas. Esta visualización permite identificar patrones operativos recurrentes y comprender cómo las diferentes cargas responden a los ciclos de ocupación y funcionamiento de la infraestructura.

Las variables Sockets plug, Lighting y Ventilation exhiben un comportamiento claramente asociado a la actividad diaria del edificio. En los tres casos se observa un incremento progresivo del consumo a partir de las 8:00 o 9:00 horas, alcanzando sus valores más altos durante la jornada laboral y disminuyendo gradualmente al finalizar el día. Adicionalmente, los fines de semana presentan niveles de consumo considerablemente inferiores a los registrados entre semana, lo que confirma que estas cargas están estrechamente relacionadas con la presencia de usuarios y el desarrollo de actividades operativas dentro de la infraestructura.

Por su parte, las variables Cooling y Heating presentan patrones temporales diferenciados. Heating registra sus mayores niveles de demanda durante las primeras horas del día, comportamiento que puede asociarse a la necesidad de acondicionar térmicamente los espacios antes o durante el inicio de las actividades. En contraste, Cooling alcanza sus máximos consumos en horas de la tarde, coincidiendo con los periodos de mayor temperatura ambiental y carga térmica interna. Estos resultados evidencian la influencia de factores climáticos sobre la demanda energética y reflejan la naturaleza complementaria de ambos sistemas de climatización.

La categoría Other electricity muestra un comportamiento distinto al de las demás variables, manteniendo

niveles de consumo relativamente elevados y estables durante las 24 horas del día y a lo largo de toda la semana. Este patrón sugiere la existencia de una carga base permanente asociada a equipos o sistemas que deben permanecer operativos de forma continua, independientemente de los horarios de ocupación. Entre estos pueden encontrarse servidores, sistemas de monitoreo, equipos de seguridad o infraestructura tecnológica crítica.

Desde una perspectiva global, los resultados permiten identificar tres componentes fundamentales del perfil energético del edificio: una carga base permanente representada principalmente por Other electricity, una carga operativa dependiente de la ocupación reflejada en Ventilation, Lighting y Sockets plug, y una carga estacional asociada a las necesidades de climatización representada por Cooling y Heating. Esta diferenciación constituye un hallazgo relevante para la caracterización del conjunto de datos, ya que demuestra que el consumo energético sigue patrones temporales predecibles y no aleatorios. Asimismo, proporciona información valiosa para las etapas posteriores de modelado, donde la incorporación de variables temporales puede contribuir significativamente a mejorar la capacidad predictiva y la precisión de los modelos de análisis de carga no intrusiva (NILM).

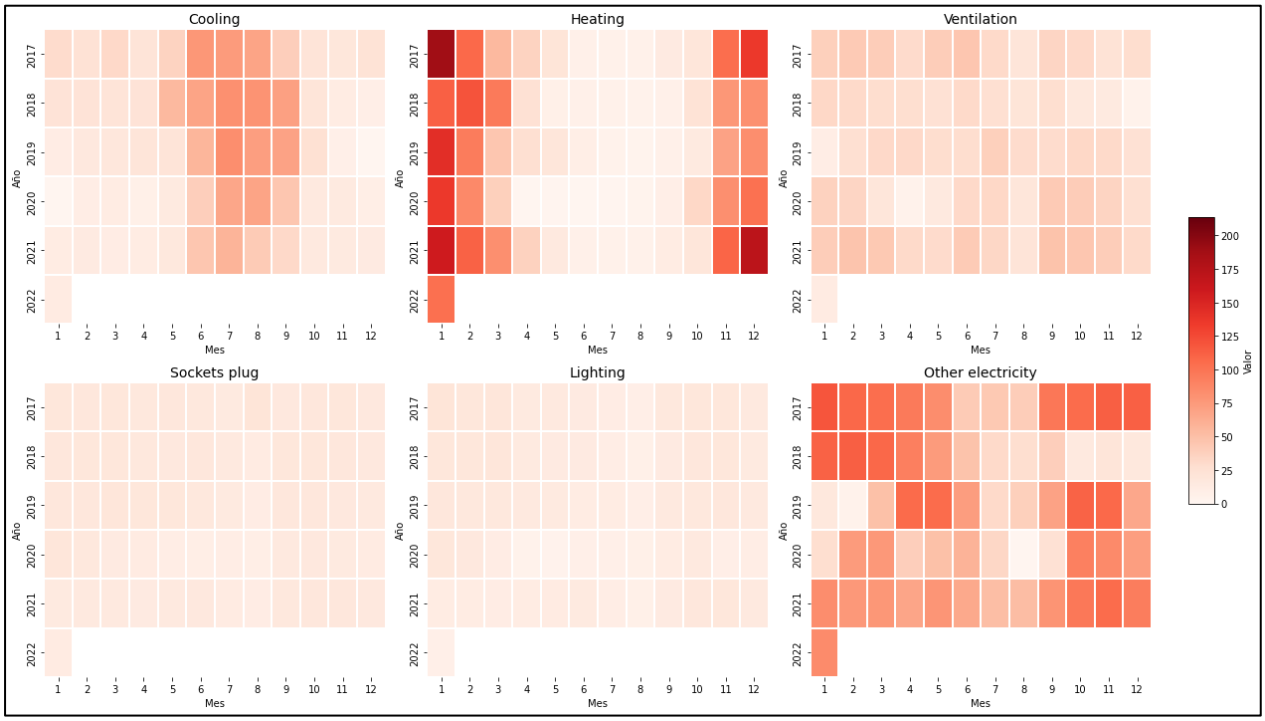


Imagen 4. Consumos por año-mes.

La Imagen 4 presenta un mapa de calor que relaciona los consumos energéticos de las variables Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating en función del mes y el año de registro. Esta representación permite identificar patrones de comportamiento de largo plazo y evaluar la influencia de factores estacionales sobre la demanda energética de la infraestructura.

Los resultados evidencian que las variables asociadas a climatización son las que presentan la mayor variabilidad temporal. En particular, Cooling concentra sus mayores niveles de consumo durante los meses

centrales del año, mientras que Heating registra incrementos significativos durante los periodos de invierno. Este comportamiento confirma la fuerte dependencia de estas cargas respecto a las condiciones climáticas y demuestra que la demanda energética del edificio está influenciada por factores ambientales externos que varían a lo largo del año. La alternancia observada entre ambas variables refleja además la naturaleza complementaria de los sistemas de calefacción y refrigeración, los cuales responden a necesidades térmicas opuestas.

Por otra parte, las variables Lighting y Sockets plug presentan una distribución relativamente homogénea durante todo el periodo de estudio. La estabilidad observada sugiere que estas cargas dependen principalmente de los patrones de ocupación y de las actividades desarrolladas en el edificio, manteniendo un comportamiento operativo consistente independientemente de la época del año. De forma similar, Ventilation muestra una variación moderada entre periodos, aunque se observa una ligera tendencia al incremento en algunos años recientes, lo que podría estar asociado a cambios en las condiciones de operación, niveles de ocupación o requerimientos de calidad del aire interior.

La variable Other electricity exhibe el comportamiento más heterogéneo entre todas las categorías analizadas. Los cambios observados entre meses y años sugieren la presencia de equipos o sistemas cuya operación no depende exclusivamente de factores climáticos ni de los horarios habituales de ocupación. Este comportamiento puede estar relacionado con modificaciones en la infraestructura tecnológica, incorporación de nuevos equipos o cambios en las necesidades operativas del edificio a lo largo del tiempo.

El análisis conjunto de las variables permite concluir que el perfil energético de la infraestructura presenta una marcada estacionalidad anual, especialmente influenciada por las necesidades de climatización. Sin embargo, también se identifican componentes de consumo relativamente estables asociados a la operación cotidiana del edificio y una carga base permanente que permanece activa durante gran parte del periodo analizado. Esta combinación de comportamientos evidencia que el consumo energético está determinado por la interacción de factores climáticos, operativos y tecnológicos.

En términos de caracterización, los resultados permiten identificar una estructura temporal multinivel. A escala anual predominan los efectos de la estacionalidad climática sobre las variables Cooling y Heating; a escala semanal y diaria, observada previamente, predominan los patrones asociados a la ocupación y funcionamiento de la infraestructura. La identificación de estos ciclos recurrentes demuestra que el consumo energético sigue patrones predecibles y no aleatorios, proporcionando información relevante para la selección de variables temporales y el desarrollo de modelos de análisis de carga no intrusiva (NILM), donde la capacidad de capturar estas dependencias temporales resulta fundamental para mejorar la precisión de la desagregación energética.

2.2.2. Limpieza de datos

A partir de los hallazgos obtenidos durante la fase de caracterización, se identificó la necesidad de realizar un proceso de limpieza y depuración del conjunto de datos con el propósito de garantizar la calidad, consistencia y confiabilidad de la información utilizada en las etapas posteriores de modelado. Aunque no se encontraron valores faltantes, se detectó la presencia de valores atípicos e inconsistencias temporales que podrían afectar la capacidad de los algoritmos para aprender patrones representativos del consumo energético.

En primer lugar, se realizó el tratamiento de los registros atípicos identificados en las variables Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating. Para ello, se empleó un método de interpolación temporal lineal, sustituyendo los valores anómalos por estimaciones calculadas a partir de observaciones

vecinas dentro de la serie temporal. Esta técnica fue seleccionada debido a que preserva la continuidad de la señal energética y mantiene la coherencia temporal de los datos, evitando introducir cambios abruptos que puedan alterar la dinámica natural del consumo.

Posteriormente, se abordaron las inconsistencias detectadas en la variable Time. Se eliminaron los registros duplicados y aquellos asociados a desfases temporales identificados durante el análisis exploratorio. Este procedimiento permitió restablecer la secuencia cronológica correcta de las observaciones y garantizar que cada registro representara una medición única dentro de la serie temporal.

Finalmente, se llevó a cabo un proceso de regularización temporal mediante la construcción de una secuencia continua con intervalos de una hora entre observaciones consecutivas. Esta etapa fue fundamental para asegurar la uniformidad temporal del conjunto de datos, requisito indispensable para la aplicación de técnicas de análisis de series temporales y algoritmos de aprendizaje automático orientados a la desagregación energética.

La aplicación de estas estrategias permitió obtener un conjunto de datos más consistente y representativo del comportamiento real de la infraestructura. En la Imagen 5. se presenta la distribución de las variables antes del proceso de limpieza, mientras que en la Imagen 6. se muestra la distribución resultante después de la depuración y regularización de los datos. La comparación entre ambas visualizaciones permite evaluar el efecto de las transformaciones realizadas y verificar la preservación de los patrones generales de consumo energético.

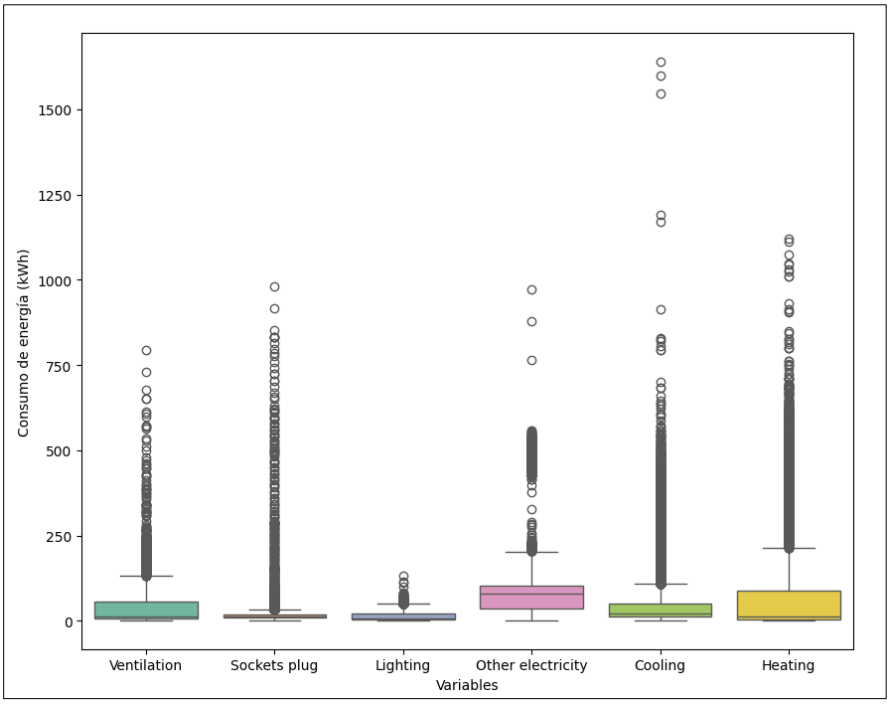


Imagen 5. Distribución de valores atípicos Set de datos original.

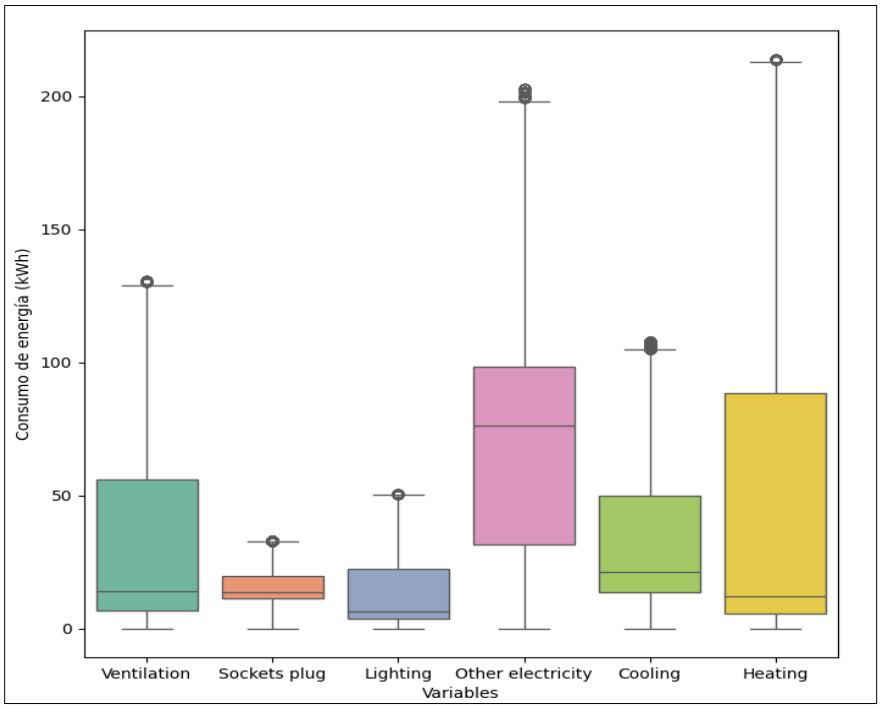


Imagen 6. Set de datos sin valores atípicos.

El procesamiento y análisis exploratorio del conjunto de datos permitió caracterizar el comportamiento del consumo energético del edificio comercial y comprender la distribución de las diferentes categorías de carga evaluadas en esta investigación. Mediante el análisis estadístico y visual se identificaron patrones temporales de consumo, relaciones entre variables, valores atípicos e inconsistencias en los registros, lo que facilitó la toma de decisiones respecto al tratamiento de la información.

Posteriormente, la etapa de limpieza permitió corregir registros inconsistentes, gestionar valores faltantes y preparar un conjunto de datos confiable para el entrenamiento de los modelos de aprendizaje automático. Estas actividades garantizaron que el proceso de modelado se desarrollara sobre información consistente y representativa del comportamiento energético real del edificio.

En conjunto, esta etapa dio cumplimiento al primer objetivo específico de la investigación, proporcionando una base metodológica sólida para la evaluación comparativa de los modelos de aprendizaje automático y para la implementación del mecanismo de reconciliación presentado en los capítulos posteriores.

3. EVALUACIÓN DE MODELOS DE MACHINE LEARNING PARA DESAGREGAR EL CONSUMO ENERGÉTICO.

3.1 Selección de modelos candidatos

A partir de la revisión del estado del arte realizada en el Capítulo 2, se identificaron los algoritmos de aprendizaje automático más utilizados en investigaciones relacionadas con la desagregación no intrusiva de cargas (NILM). La Tabla 5 presenta la frecuencia de utilización de diferentes modelos en los trabajos analizados, evidenciando una mayor presencia de Redes Neuronales Recurrentes con arquitectura LSTM (RNN-LSTM), Bosques Aleatorios (Random Forest), Máquinas de Soporte Vectorial (SVM) y XGBoost.

Con base en estos resultados, se seleccionaron cinco modelos de regresión para su implementación y evaluación en esta investigación: Regresión Lineal, Máquinas de Soporte Vectorial (SVM), Bosques Aleatorios (Random Forest), XGBoost (Extreme Gradient Boosting) y Redes Neuronales Recurrentes (RNN-LSTM). La selección buscó incluir algoritmos con diferentes niveles de complejidad y mecanismos de aprendizaje, permitiendo comparar enfoques lineales, métodos basados en márgenes, técnicas de aprendizaje por conjuntos y arquitecturas de aprendizaje profundo.

La Regresión Lineal fue utilizada como modelo base de referencia debido a su simplicidad e interpretabilidad. Los modelos SVM, Random Forest y XGBoost fueron seleccionados por su amplio uso en problemas de regresión no lineal y por los resultados reportados en aplicaciones NILM. Finalmente, las redes RNN-LSTM fueron incluidas debido a su capacidad para modelar dependencias temporales, característica especialmente relevante en series de consumo energético donde existen patrones horarios, semanales y estacionales identificados durante la fase de caracterización.

Tabla 5. Cantidad de modelos por referencias

Modelo	[3]	[9]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]	[20]	[21]	Total
RNN-LSTM		x	x	x	x	x	x		x	x	x		9
Random Forest		x	x		x							x	4
SVM				x	x	x							3
XGBoost		x			x		x						3
Regresión Lineal	x												1
KNN								x					1

Para cuantificar el desempeño de los modelos se emplearon el error cuadrático medio (RMSE), el error absoluto medio (MAE) y el coeficiente de determinación (R^2), cuyas definiciones formales se presentan en la Sección 1.3 del marco teórico. Estas métricas se evaluaron sobre el conjunto de entrenamiento y el de prueba, con el fin de estimar la capacidad de ajuste y generalización de cada modelo. Adicionalmente, se incorporó el indicador Gap R, definido como la diferencia entre el R^2 de entrenamiento y el R^2 de prueba, el cual cuantifica el nivel de sobreajuste de cada configuración evaluada.

3.2 Preparación y preprocesamiento de datos para el modelado

Una vez finalizado el proceso de limpieza y validación de la calidad de los datos, se realizó la preparación de las variables necesarias para el entrenamiento de los modelos de Machine Learning. El objetivo de esta etapa fue construir un conjunto de características capaz de representar tanto el comportamiento temporal del consumo energético como la información agregada disponible en un escenario de desagregación no intrusiva de cargas (NILM).

La Tabla 6 presenta la clasificación de las variables utilizadas durante el proceso de modelado. Las variables objetivo corresponden a Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating, las cuales representan el consumo energético desagregado de cada categoría de carga en kWh. Estas variables constituyen las salidas que los modelos deben estimar a partir de la información disponible en las variables predictoras.

Como variable principal de entrada se empleó Total, calculada como la suma de los consumos de Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating. Esta variable representa el consumo energético agregado del edificio y permite reproducir el escenario característico de la desagregación no intrusiva de cargas, donde el objetivo consiste en estimar el aporte individual de cada carga a partir de una medición global.

Adicionalmente, se generaron variables temporales derivadas de la marca de tiempo original (Time), incluyendo Hora, Día, Mes, Año y DiaSemanaNum. Estas variables fueron obtenidas mediante la descomposición de la fecha y hora de registro con el fin de capturar los patrones temporales identificados durante la fase de caracterización. La variable Hora permite modelar los ciclos diarios de ocupación, DiaSemanaNum representa las diferencias entre días laborables y fines de semana, mientras que Mes y Año incorporan información relacionada con la estacionalidad y la evolución temporal del consumo energético.

Debido a que algunos algoritmos son sensibles a las diferencias de escala entre las variables, se aplicaron procedimientos de escalamiento específicos según el modelo utilizado. Para las Máquinas de Soporte Vectorial (SVM) se empleó StandardScaler sobre las variables predictoras, transformando los datos para que presentaran media cero y desviación estándar unitaria. Esta transformación favorece el cálculo de distancias y mejora la estabilidad del proceso de optimización.

Por otra parte, para el modelo RNN-LSTM se utilizó MinMaxScaler tanto en las variables de entrada como en las variables objetivo, escalando los valores al intervalo [0,1]. Este procedimiento facilita el entrenamiento de redes neuronales recurrentes al evitar que diferencias significativas de magnitud entre variables afecten el proceso de aprendizaje y la actualización de los pesos de la red. Una vez finalizado el entrenamiento, las predicciones fueron transformadas nuevamente a la escala original para realizar la evaluación de desempeño mediante las métricas definidas.

Como resultado de esta etapa se obtuvo un conjunto de datos estructurado y preparado para el entrenamiento y evaluación de los diferentes modelos de regresión considerados en la investigación

Tabla 6. Clasificación de variables

Variable	Tipo	Clasificación variable	Observación
Time	Original	Predictora	Variable de fecha y hora.
Ventilation	Original	Objetivo	Representan el consumo de energía en kWh de la variable ventilación.
Socket plug	Original	Objetivo	Representan el consumo de energía en kWh de la variable enchufes.
Lighting	Original	Objetivo	Representan el consumo de energía en kWh de la variable iluminación.
Other electricity	Original	Objetivo	Representan el consumo de energía en kWh de la variable otros equipos.
Cooling	Original	Objetivo	Representan el consumo de energía en kWh de la variable enfriamiento.
Heating	Original	Objetivo	Representan el consumo de energía en kWh de la variable calefacción.
Hora	Calculada	Predictora	Representa la hora, con valores de 1 a 24, calculada de la variable Time
Dia	Calculada	Predictora	Representa el numero de día del mes, con valores de 1 a 31, calculada de la variable Time
Mes	Calculada	Predictora	Representa el mes, con valores de 1 a 12, calculada de la variable Time
Anno	Calculada	Predictora	Representa el Año, con valores de 2017–2022, calculada de la variable Time
DiaSemanaNum	Calculada	Predictora	Representa el día de la semana, con valores de 1 a 7, calculada de la variable Time
Total	Calculada	Predictora	Representa el total de las suma de las variables "Ventilation","Sokets plug", "Lighting", "Other electricity", "Cooling", "Heating"

3.3 Diseño experimental y partición de datos

Con el propósito de evaluar la capacidad de generalización de los modelos propuestos, el conjunto de datos fue dividido en dos subconjuntos: entrenamiento y prueba. En todos los casos se destinó el 80 % de las observaciones para el entrenamiento de los modelos y el 20 % restante para la evaluación de desempeño. Esta estrategia permite ajustar los algoritmos utilizando una muestra representativa de los datos disponibles y posteriormente validar su capacidad de predicción sobre observaciones no empleadas durante el proceso de aprendizaje.

La estrategia de partición fue adaptada según las características de cada algoritmo. Para los modelos de Regresión Lineal, Máquinas de Soporte Vectorial (SVM), XGBoost y RNN-LSTM se utilizó una división secuencial de la serie temporal, donde los primeros registros fueron destinados al entrenamiento y los registros más recientes se reservaron para la fase de prueba. Esta aproximación permite mantener la estructura temporal de los datos y evaluar el comportamiento de los modelos en condiciones similares a un escenario real de predicción.

Por otra parte, para el modelo Random Forest se empleó una partición aleatoria mediante la función train_test_split(), utilizando un 20 % de los datos para prueba y una semilla de aleatoriedad fija (random_state = 42). Esta estrategia fue seleccionada debido a la naturaleza del algoritmo, basado en

árboles de decisión y métodos de ensamble, donde la aleatorización constituye un componente inherente de su proceso de aprendizaje.

Una vez realizada la partición, cada modelo fue entrenado utilizando exclusivamente los datos correspondientes al conjunto de entrenamiento. Posteriormente, las predicciones generadas sobre el conjunto de prueba fueron empleadas para calcular las métricas de desempeño definidas en la investigación, permitiendo comparar objetivamente la capacidad de desagregación energética de cada algoritmo.

3.4 Optimización de hiperparámetros

Con el objetivo de maximizar el desempeño predictivo de los modelos evaluados, se implementó un proceso de optimización de hiperparámetros para las Máquinas de Soporte Vectorial (SVM), Bosques Aleatorios (Random Forest), XGBoost (Extreme Gradient Boosting) y Redes Neuronales Recurrentes (RNN-LSTM). La Regresión Lineal fue utilizada como modelo base de referencia y, debido a su naturaleza paramétrica, no requirió ajuste de hiperparámetros.
La optimización se desarrolló mediante una estrategia iterativa compuesta por dos etapas. Inicialmente, se aplicó RandomizedSearchCV con el propósito de explorar amplias regiones del espacio de búsqueda definido para cada algoritmo. Esta fase permitió identificar configuraciones prometedoras reduciendo el costo computacional asociado a la evaluación exhaustiva de todas las combinaciones posibles. Posteriormente, a partir de los mejores resultados obtenidos, se definieron rangos de búsqueda más acotados sobre los cuales se ejecutó Grid Search, permitiendo refinar la selección de hiperparámetros y obtener configuraciones más precisas.
La Tabla 7 presenta el espacio de búsqueda definido para cada modelo. En el caso de las Máquinas de Soporte Vectorial (SVM), se evaluaron diferentes configuraciones del núcleo de transformación (kernel), el parámetro de regularización (C), el margen de tolerancia (epsilon) y el parámetro gamma. Para los Bosques Aleatorios (Random Forest) se analizaron diferentes combinaciones relacionadas con el número de árboles, profundidad máxima, tamaño mínimo de división, tamaño mínimo de hoja y cantidad de características utilizadas durante la construcción de cada árbol.
En el modelo XGBoost se optimizaron parámetros asociados al proceso de boosting, incluyendo el número de estimadores, profundidad máxima de los árboles, tasa de aprendizaje, fracciones de muestreo de observaciones y características, así como mecanismos de regularización orientados a controlar el sobreajuste. Por su parte, para la arquitectura RNN-LSTM se evaluaron diferentes configuraciones relacionadas con la estructura de la red neuronal, incluyendo el número de unidades ocultas, cantidad de capas recurrentes, tamaño de ventana temporal, tasa de aprendizaje, tamaño de lote, número de épocas de entrenamiento y mecanismos de regularización mediante dropout y parada temprana.
El proceso de optimización fue ejecutado de forma independiente para cada modelo, utilizando exclusivamente el conjunto de entrenamiento definido en la fase experimental. Posteriormente, las configuraciones con mejor desempeño fueron seleccionadas para la evaluación final sobre el conjunto de prueba. Esta estrategia permitió comparar los modelos bajo condiciones optimizadas y garantizar que los resultados obtenidos reflejaran el potencial de cada algoritmo para la desagregación no intrusiva del consumo energético.

Tabla 7. Espacio de hiperparámetros fase de búsqueda

Algoritmo	Hiperparametros	Espacio búsqueda
Regresión Lineal	-	Sin busqueda
SVM	kernel	[rbf, linear]
SVM	C	[1, 10, 50, 100, 200]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]
RandomForest	n_estimators	[100, 200, 300, 400, 500, 600]
RandomForest	max_depth	[None, 5, 10 , 15, 20]
RandomForest	min_samples_split	[2, 5, 7, 10]
RandomForest	min_samples_leaf	[1, 2, 4, 6]
RandomForest	max_features	[log2, sqrt]
RandomForest	max_samples	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]
XGBoost	max_depth	[4, 6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]
XGBoost	subsample	[0,8]
XGBoost	colsample_bytree	[0.8, 1]
XGBoost	min_child_weight	[1, 5]
XGBoost	gamma	0
XGBoost	reg_lambda	1
XGBoost	reg_alpha	0
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]
RNN-lstm	window_size	[24, 48, 72, 168, 336]
RNN-lstm	activation_function	[relu, tanh, elu]
RNN-lstm	batch_size	[12, 32, 64, 128]
RNN-lstm	loss	mse
RNN-lstm	monitor	val_loss
RNN-lstm	patience	5

- **Regresión Lineal:** no requiere ajuste de hiperparámetros
- **Máquinas de Soporte Vectorial (SVM)**: núcleo (kernel), parámetro de penalización (C), ancho de la zona de tolerancia ε-insensitive (epsilon), coeficiente del núcleo (gamma).

- **Bosques Aleatorios (Random Forest):** número de árboles (n_estimators), profundidad máxima (max_depth), mínimo de muestras para dividir (min_samples_split), mínimo de muestras en una hoja (min_samples_leaf), máximo de características (max_features), máximo de muestras por árbol (max_samples).

- **XGBoost (Extreme Gradient Boosting):** número de árboles / iteraciones de boosting (n_estimators), profundidad máxima por árbol (max_depth), tasa de aprendizaje (learning_rate), fracción de muestras por árbol (subsample), fracción de características por árbol (colsample_bytree), peso mínimo de instancia por hoja (min_child_weight), reducción mínima de pérdida para dividir (gamma), regularización L2 (reg_lambda), regularización L1 (reg_alpha).

- **Redes Neuronales Recurrentes (RNN-LSTM):** número de unidades ocultas por capa (units o hidden_units), número de capas LSTM (num_layers), tamaño de la ventana temporal o pasos de tiempo de entrada (window_size), tasa de aprendizaje del optimizador (learning_rate), tamaño del lote de entrenamiento (batch_size), número de épocas de entrenamiento (epochs), función de activación de la LSTM (activation), optimizador (optimizer), porcentaje de desactivación de neuronas para regularización (dropout), tipo de función de pérdida (loss), número de neuronas en la capa de salida (output_units), tipo de capa de salida (Dense), criterio de parada temprana (early stopping: patience, monitor).

3.5 Flujo metodológico para el entrenamiento y evaluación de modelos

La Imagen 7 presenta el flujo metodológico implementado para el desarrollo, optimización y evaluación de los modelos de Machine Learning utilizados en la investigación. El proceso inicia con la carga del conjunto de datos previamente depurado y la preparación de las variables predictoras y objetivo definidas en la etapa de preprocesamiento.

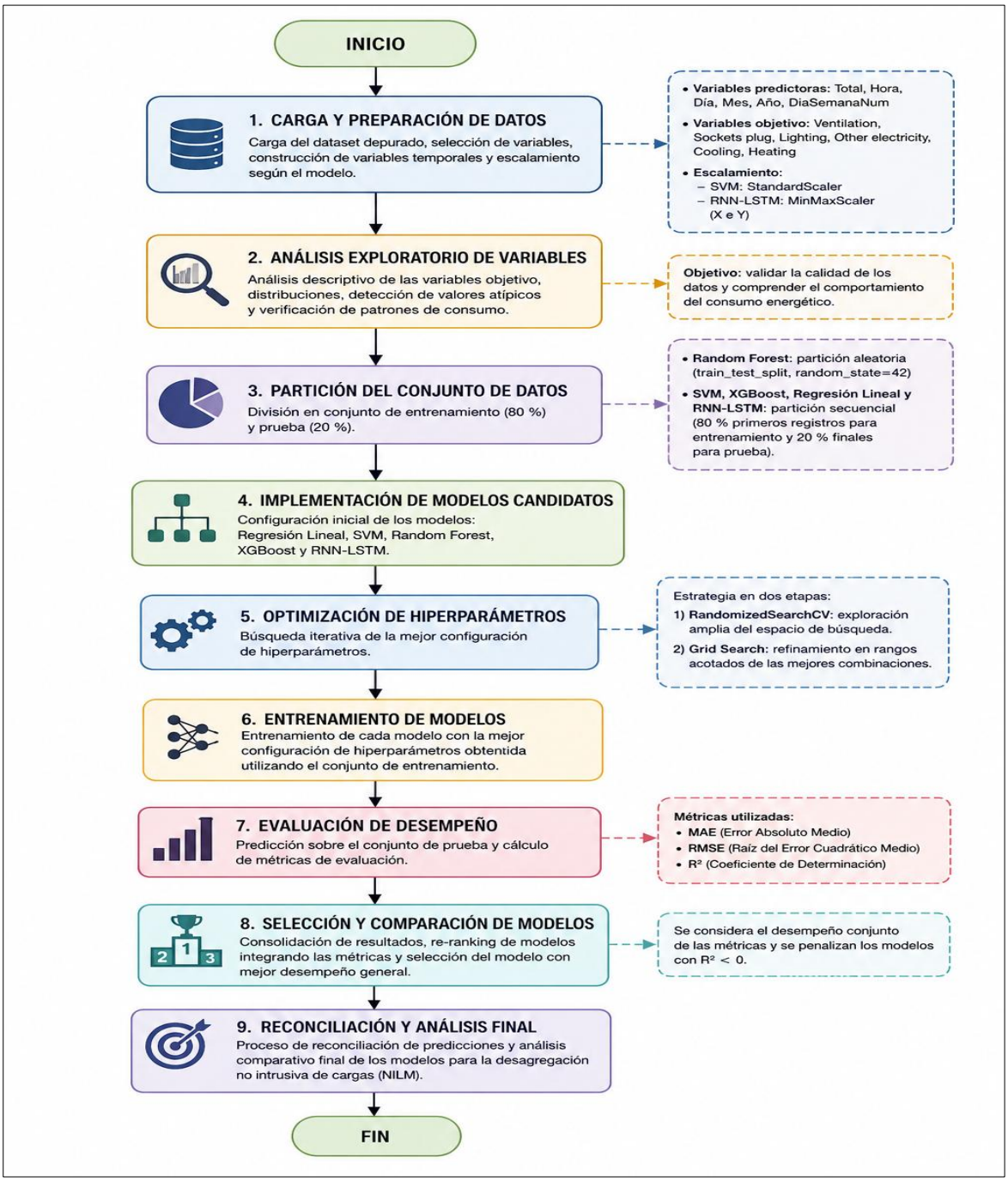
Posteriormente, se realiza una exploración descriptiva de las variables de salida con el propósito de verificar su distribución, identificar posibles comportamientos atípicos y validar la consistencia de los datos antes del entrenamiento. Esta etapa permitió corroborar que las transformaciones y procedimientos de limpieza aplicados durante la caracterización preservaron los patrones de consumo energético presentes en el conjunto de datos.

Una vez validada la calidad de la información, se lleva a cabo la partición de los datos en conjuntos de entrenamiento y prueba utilizando una proporción de 80 % y 20 %, respectivamente. Esta división permite entrenar los modelos utilizando una muestra representativa de los datos disponibles y evaluar posteriormente su capacidad de generalización sobre observaciones no utilizadas durante el aprendizaje.

A continuación, se implementan los modelos candidatos seleccionados en la investigación: Regresión Lineal, Máquinas de Soporte Vectorial (SVM), Bosques Aleatorios (Random Forest), XGBoost (Extreme Gradient Boosting) y Redes Neuronales Recurrentes (RNN-LSTM). Cada algoritmo es configurado de acuerdo con sus requerimientos específicos y preparado para el proceso de optimización de hiperparámetros.

La optimización se desarrolla mediante una estrategia iterativa que combina RandomizedSearchCV y Grid Search. Inicialmente, se exploran amplias regiones del espacio de búsqueda para identificar configuraciones prometedoras. Posteriormente, se refinan los rangos de búsqueda alrededor de las mejores combinaciones encontradas, permitiendo obtener configuraciones más precisas y reducir el riesgo de seleccionar parámetros subóptimos.

Imagen 7. Flujo metodológico para el entrenamiento.



Finalizado el entrenamiento, cada modelo genera predicciones sobre el conjunto de prueba. Estas predicciones son evaluadas mediante las métricas MAE, RMSE y coeficiente de determinación R^2, permitiendo cuantificar la precisión y capacidad explicativa de cada algoritmo. Los resultados obtenidos

son posteriormente consolidados y comparados mediante un proceso de clasificación de desempeño, considerando simultáneamente las diferentes métricas evaluadas.

Con base en los resultados obtenidos, se identifican las configuraciones de hiperparámetros con mejor desempeño para cada modelo. Esta información es utilizada para realizar nuevas iteraciones de optimización cuando es necesario, ajustando progresivamente los espacios de búsqueda con el fin de mejorar la precisión predictiva y reducir el costo computacional asociado al entrenamiento.

Finalmente, los resultados de todos los modelos son integrados en una etapa de reconciliación y comparación global, permitiendo identificar la alternativa con mejor desempeño para la tarea de desagregación no intrusiva del consumo energético. Los resultados detallados de este proceso y el análisis comparativo de los modelos se presentan en el capítulo siguiente.

4. RESULTADOS

En el análisis comparativo del desempeño de distintos modelos de regresión aplicados a la estimación de variables de consumo energético desagregado, específicamente Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating, se evaluaron cinco enfoques de modelado: Random Forest, XGBoost, SVM, redes neuronales tipo RNN-LSTM y Regresión Lineal. Cada modelo fue sometido a tres fases de configuración, excepto el modelo de Regresión Lineal, que únicamente se evaluó en la fase de búsqueda: búsqueda inicial de hiperparámetros (Búsqueda), ajuste fino (Optimizado) y una segunda iteración de refinamiento (Optimizado 2), con el fin de identificar configuraciones capaces de maximizar el desempeño predictivo.

El desempeño de los modelos se midió mediante métricas estándar de regresión: error absoluto medio (MAE), raíz del error cuadrático medio (RMSE) y coeficiente de determinación (R^2), evaluado tanto en el conjunto de prueba como en entrenamiento (R^2 Train). Estas métricas permiten analizar no solo la precisión predictiva, sino también la capacidad de generalización del modelo. Adicionalmente, se incorporó el indicador Gap R, definido como la diferencia entre el R^2 de entrenamiento y el R^2 de prueba, el cual permite cuantificar el nivel de sobreajuste o estabilidad de cada configuración; valores bajos indican una mejor capacidad de generalización, mientras que diferencias elevadas sugieren que el modelo aprende excesivamente los patrones del conjunto de entrenamiento sin transferir adecuadamente dicho aprendizaje a datos no observados.

Con el propósito de sintetizar el análisis y facilitar la comparación global entre modelos, se construyó un indicador de ranking (Rank) que ordena las configuraciones según su desempeño priorizando altos valores de R^2 en prueba. De esta manera, el ranking permite identificar los modelos más precisos. Los resultados obtenidos se presentan en la Tabla 21. Resultados de desempeño de modelos de regresiónEn Anexos

4.1 Resultados de la variable Lighting

La interpretación de los resultados se fundamenta en la información presentada en las Tabla 8 y Tabla 9, donde se resumen tanto el desempeño de los modelos mediante métricas de error (MAE, RMSE y R^2) como las configuraciones de hiperparámetros evaluadas en cada fase. A partir de estas tablas, se establece la relación entre la optimización de los modelos y su capacidad predictiva, permitiendo identificar las

configuraciones más eficientes y analizar el equilibrio entre precisión y generalización.

- En Random Forest, se observa una mejora consistente a medida que se incrementa la complejidad del modelo. El aumento de n_estimators (de 100 a 600) y la flexibilización de max_depth (hasta None) permiten reducir el MAE de 2,9 a 1,8 y el RMSE de 4,4 a 2,7, mientras que el R^2 Test aumenta de 90,6% a 96,5%. Esta mejora simultánea en métricas de error y ajuste confirma que el modelo logra una representación más precisa de los datos. Sin embargo, el incremento del Gap R (de 0,7 a 3) indica un leve compromiso en la generalización, asociado al aumento de la complejidad.

- Para SVM, la optimización de los hiperparámetros (C, epsilon y gamma) también genera mejoras importantes. El MAE disminuye de 3,3 a 1,9 y el RMSE de 5,1 a 3, mientras que el R^2 Test alcanza 95,7%. A diferencia de Random Forest, el Gap R se mantiene bajo (≈2), lo que evidencia un mejor equilibrio entre precisión y generalización. Esto sugiere que SVM logra reducir el error sin incurrir en un sobreajuste significativo.

Tabla 8. Desempeño de los modelos en la variable Lighting.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Lighting	1,8	2,7	96,5	99,5	3	1
SVM	Optimizado 2	Lighting	1,9	3	95,7	97,6	2	3
Random Forest	Optimizado	Lighting	2	3,1	95,4	97,4	2,1	5
XGBoost	Optimizado 2	Lighting	1,8	3,1	93,7	98,6	4,9	8
SVM	Optimizado	Lighting	2,3	3,6	93,6	95,2	1,6	9
Random Forest	Búsqueda	Lighting	2,9	4,4	90,6	91,4	0,7	16
XGBoost	Optimizado	Lighting	3,1	4,1	89,1	92,2	3,1	19
SVM	Búsqueda	Lighting	3,3	5,1	87,2	88	0,8	22
XGBoost	Búsqueda	Lighting	3,7	4,8	85,4	88,1	2,7	28
RNN-lstm	Optimizado 2	Lighting	3,9	5,9	78	86,1	8,1	43
RNN-lstm	Optimizado	Lighting	4,2	6,4	73,6	92,9	19,3	52
RNN-lstm	Búsqueda	Lighting	4,4	6,4	73,5	92,4	18,9	53
Regresión Lineal	Búsqueda	Lighting	6,6	9,6	35,1	33,7	1,5	71

- En el caso de XGBoost, la optimización produce una reducción del error (MAE de 3,7 a 1,8 y RMSE de 4,8 a 3,1) y un aumento del R^2 Test hasta 93,7%. No obstante, estos beneficios vienen acompañados de un incremento en el Gap R (hasta 4,9), lo que indica que parte de la mejora en precisión proviene de un mayor ajuste al conjunto de entrenamiento, especialmente influenciado por el incremento del learning_rate.

Tabla 9. Configuraciones de hiperparámetros evaluadas en la variable Lighting.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	100	400	600	[100, 400, 600]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	10	15	None	[None, 10 , 15]
Random Forest	min_samples_split	[2, 5, 7, 10]	5	2	2	[2, 5]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	1	1	1	[1]
Random Forest	max_features	[log2, sqrt]	log2	sqrt	sqrt	[log2, sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	400	300	500	[300, 400, 500]
XGBoost	max_depth	[4, 6]	6	6	6	[6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,003	0,005	0,05	[0.003, 0.005, 0.05]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	1	1	0,8	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	1	1	5	[1, 5]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	100	10	100	[10, 100]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,01	0,05	0,2	[0.01, 0.05, 0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	scale	1	1	[scale, 1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,5	0,5	0,5	[0.5]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,001	0,003	0,003	[0.001, 0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(128,64,32)	(128,64,32)	(64,32,16)	[(128,64,32), (64,32,16)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Adam	Adam	Rmsprop	[adam, rmsprop]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	336	336	336	[336]
RNN-lstm	activation_function	[relu, tanh, elu]	Relu	Relu	Tanh	[relu, tanh]
RNN-lstm	batch_size	[12, 32, 64, 128]	64	64	64	[64]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- Por otro lado, los modelos RNN-LSTM muestran una reducción moderada en las métricas de error (MAE de 4,4 a 3,9), pero sin mejoras significativas en el R^2 Test (alrededor de 73%–78%). Además, presentan valores elevados de RMSE (hasta 6,4) y Gap R muy altos (hasta 19,3), lo que evidencia una alta varianza y problemas de sobreajuste. Esto sugiere que, aunque el modelo logra cierto ajuste en términos de error, no generaliza adecuadamente.

- Finalmente, la Regresión Lineal presenta los peores resultados, con los mayores valores de error (MAE = 6,6; RMSE = 9,6) y un R^2 Test de 35,1%, confirmando que un modelo lineal es insuficiente para capturar la complejidad de la variable Lighting.

4.2 Resultados de la variable Heating

El análisis conjunto de las métricas de error (MAE, RMSE), el coeficiente de determinación (R^2) y la configuración de hiperparámetros de la Tabla 10 y Tabla 11 respectivamente, permite evidenciar cómo la

optimización impacta el desempeño de los modelos en la variable Heating.

Tabla 10. Desempeño de los modelos en la variable Heating.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Heating	6,7	12,9	96,1	99,5	3,4	2
SVM	Optimizado 2	Heating	6,8	14	95,5	97	1,5	4
XGBoost	Optimizado 2	Heating	8,8	15,7	95,1	99,2	4	6
Random Forest	Optimizado	Heating	9,3	16,3	93,8	95,8	2	7
XGBoost	Optimizado	Heating	12	19,3	92,6	95,6	3	10
SVM	Optimizado	Heating	10,2	18,6	92	93,2	1,2	12
Random Forest	Búsqueda	Heating	11,9	19	91,6	93,1	1,4	13
XGBoost	Búsqueda	Heating	17,7	24	88,6	91,9	3,3	20
SVM	Búsqueda	Heating	14,1	24,7	85,8	86,5	0,6	26
RNN-lstm	Optimizado 2	Heating	25,1	46,6	83	95	12	31
RNN-lstm	Optimizado	Heating	26,4	47,7	82	93	11	34
RNN-lstm	Búsqueda	Heating	27,9	51,3	79	93	14	41
Regresión Lineal	Búsqueda	Heating	33,2	44,5	57,7	59,6	1,9	63

- En Random Forest, la mejora del modelo está directamente asociada al incremento en la complejidad estructural, particularmente mediante el aumento de n_estimators (de 400 a 600) y la eliminación de restricciones en la profundidad (max_depth = None). Esto permite reducir el MAE de 11,9 a 6,7 y el RMSE de 19 a 12,9, mientras que el R^2 Test se incrementa de 91,6% a 96,1%. Sin embargo, el Gap R también aumenta (de 1,4 a 3,4), lo que indica un ligero sobreajuste, aunque dentro de niveles aceptables considerando la ganancia en precisión.

- Para SVM, la optimización de los hiperparámetros (C, epsilon y gamma) muestra un impacto significativo en la reducción del error. El MAE disminuye de 14,1 a 6,8 y el RMSE de 24,7 a 14, mientras que el R^2 Test alcanza 95,5%. A diferencia de Random Forest, el Gap R se mantiene bajo (1,5), lo que evidencia una mejor capacidad de generalización. Esto confirma que SVM logra un balance adecuado entre complejidad y estabilidad del modelo.

Tabla 11. Configuraciones de hiperparámetros evaluadas en la variable Heating.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	400	400	600	[400, 600]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	10	20	None	[None, 10 , 20]
Random Forest	min_samples_split	[2, 5, 7, 10]	2	10	2	[2, 10]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	2	6	1	[1, 2, 6]
Random Forest	max_features	[log2, sqrt]	log2	log2	sqrt	[log2, sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	600	600	600	[600]
XGBoost	max_depth	[4, 6]	6	6	6	[6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,003	0,005	0,05	[0.003, 0.005, 0.05]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	1	1	0,8	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	1	1	1	[1]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	100	10	100	[10, 100]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,01	0,2	0,2	[0.01, 0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	scale	1	1	[scale, 1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,5	0,5	0,5	[0.5]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,003	0,003	0,003	[0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(64,32,16)	(64,32,16)	(64,32,16)	[(64,32,16)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Adam	Adam	Adam	[adam]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	336	336	168	[168, 336]
RNN-lstm	activation_function	[relu, tanh, elu]	Tanh	Relu	Relu	[relu, tanh]
RNN-lstm	batch_size	[12, 32, 64, 128]	64	32	32	[3, 64]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- En el caso de XGBoost, el aumento del learning_rate (de 0,003 a 0,05) y el uso de un número elevado de estimadores (600) permiten mejorar el R^2 Test de 88,6% a 95,1%, acompañado de una reducción del MAE (de 17,7 a 8,8) y del RMSE (de 24 a 15,7). No obstante, el Gap R se incrementa hasta 4, lo que indica una mayor tendencia al sobreajuste, similar a lo observado en otras variables.

- Por otro lado, los modelos RNN-LSTM presentan mejoras marginales en las métricas de error (MAE de 27,9 a 25,1), pero mantienen valores elevados de RMSE (superiores a 46) y un R^2 Test limitado (máximo de 83%). Además, el Gap R es considerablemente alto (entre 11 y 14), lo que evidencia un fuerte sobreajuste. A pesar de ajustes en hiperparámetros como window_size, arquitectura de capas y función de activación, el modelo no logra capturar eficientemente la dinámica de la variable.

- Finalmente, la Regresión Lineal presenta el menor desempeño, con un MAE de 33,2, RMSE de 44,5 y un R^2 Test de 57,7%, lo que confirma que las relaciones en la variable Heating no son adecuadamente representadas mediante modelos lineales.

4.3 Resultados de la variable Socket Plug

El análisis integrado de las métricas de desempeño y las configuraciones de hiperparámetros de la Tabla 12 y Tabla 13 respectivamente, permite identificar cómo los procesos de ajuste influyen en la precisión y capacidad de generalización de los modelos para la variable Sockets plug, evidenciando diferencias importantes en su comportamiento respecto a otras variables.

Tabla 12. Desempeño de los modelos en la variable Socket Plug.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Sockets plug	1,4	1,9	92,2	98,9	6,7	11
Random Forest	Optimizado	Sockets plug	1,5	2,1	90,8	95,7	4,8	15
SVM	Optimizado 2	Sockets plug	1,5	2,2	89,6	94,4	4,8	18
SVM	Optimizado	Sockets plug	1,7	2,6	86,2	89,7	3,5	25
Random Forest	Búsqueda	Sockets plug	2	2,9	82,7	84,5	1,8	32
SVM	Búsqueda	Sockets plug	2	3,1	79,6	82	2,3	38
XGBoost	Optimizado 2	Sockets plug	2	3,1	79,2	95,3	16,1	40
XGBoost	Optimizado	Sockets plug	2,1	3,3	76,9	95,4	18,5	46
XGBoost	Búsqueda	Sockets plug	2,3	3,4	74,6	90,9	16,2	51
RNN-lstm	Búsqueda	Sockets plug	3,3	4,7	51	81,5	30,5	68
RNN-lstm	Optimizado 2	Sockets plug	3,4	4,9	48,3	58,6	10,4	69
RNN-lstm	Optimizado	Sockets plug	3,4	2	45,2	78,5	33,3	70
Regresión Lineal	Búsqueda	Sockets plug	3,9	5,3	28,1	27,8	0,3	72

- En Random Forest, la optimización muestra mejoras claras en todas las métricas. El ajuste de n_estimators (de 100 a 300), junto con el incremento en la profundidad (max_depth) y el cambio en max_features hacia sqrt, permite reducir el MAE de 2 a 1,4 y el RMSE de 2,9 a 1,9, mientras que el R^2 Test aumenta de 82,7% a 92,2%. Sin embargo, el Gap R crece hasta 6,7, lo que indica un mayor nivel de sobreajuste en comparación con otras variables, sugiriendo que el modelo se vuelve más sensible a los datos de entrenamiento.

- Para SVM, la optimización de los hiperparámetros (C, epsilon y gamma) también genera mejoras consistentes. El MAE disminuye de 2 a 1,5 y el RMSE de 3,1 a 2,2, alcanzando un R^2 Test de 89,6%. El Gap R se mantiene en niveles moderados (4,8), lo que indica un balance aceptable entre

precisión y generalización, aunque ligeramente inferior al observado en variables como Lighting o Heating.

Tabla 13. Configuraciones de hiperparámetros evaluadas en la variable Socket Plug.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	100	300	300	[100, 300]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	10	15	None	[None, 10 , 15]
Random Forest	min_samples_split	[2, 5, 7, 10]	5	2	2	[2, 5]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	1	1	1	[1]
Random Forest	max_features	[log2, sqrt]	log2	sqrt	sqrt	[log2, sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	400	200	600	[200, 400, 500]
XGBoost	max_depth	[4, 6]	4	6	4	[4, 6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,005	0,03	0,05	[0.005, 0.03, 0.05]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	1	1	0,8	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	5	1	5	[1, 5]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	1	10	10	[1, 10]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,2	0,2	0,2	[0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	1	1	1	[1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,2	0,5	0,5	[0,2, 0.5]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,003	0,003	0,003	[0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(128,64,32)	(128,64,32)	(128,64,32)	[(128,64,32)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Adam	Adam	Rmsprop	[adam, rmsprop]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	168	168	336	[168, 336]
RNN-lstm	activation_function	[relu, tanh, elu]	Relu	Relu	Relu	[relu]
RNN-lstm	batch_size	[12, 32, 64, 128]	64	64	64	[64]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- En el caso de XGBoost, aunque se observa una reducción moderada en las métricas de error (MAE de 2,3 a 2 y RMSE de 3,4 a 3,1), el R^2 Test solo alcanza 79,2%. Además, el Gap R es considerablemente alto (superior a 16 en todas las fases), lo que evidencia un fuerte sobreajuste. Este comportamiento está asociado al uso de tasas de aprendizaje más altas (learning_rate) y un número elevado de estimadores, lo que incrementa la complejidad del modelo sin traducirse en mejoras proporcionales en el desempeño en prueba.

- Por otro lado, los modelos RNN-LSTM presentan un desempeño bajo, con valores de R^2 Test entre 45% y 51% y errores elevados (MAE superiores a 3,3). A pesar de ajustes en hiperparámetros como dropout, window_size y arquitectura de capas, el modelo no logra capturar adecuadamente los

patrones de la variable. Adicionalmente, los valores de Gap R son extremadamente altos (hasta 33,3), lo que confirma problemas severos de sobreajuste.

- Finalmente, la Regresión Lineal presenta el peor desempeño, con un R^2 Test de 28,1% y los mayores valores de error (MAE = 3,9; RMSE = 5,3), lo que evidencia que la variable Sockets plug posee una estructura no lineal que no puede ser capturada mediante modelos simples.

4.4 Resultados de la variable Cooling

Para la variable Cooling, el análisis integrado de las métricas de error (MAE y RMSE), el coeficiente de determinación (R^2) y la configuración de hiperparámetros de la Tabla 14 y Tabla 15 respectivamente, evidencia cómo los procesos de optimización influyen en el desempeño de los modelos.

Tabla 14. Desempeño de los modelos en la variable Cooling.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Cooling	5	8,3	91,1	98,8	7,7	14
Random Forest	Optimizado	Cooling	5,2	8,5	90,6	96,1	5,5	17
XGBoost	Optimizado 2	Cooling	4,9	7,4	87,9	94,6	6,7	21
SVM	Optimizado 2	Cooling	6,1	10,1	86,7	90,1	3,4	23
XGBoost	Optimizado	Cooling	5,6	7,8	86,4	90,5	4,1	24
SVM	Optimizado	Cooling	6,4	10,5	85,7	88,7	3	27
Random Forest	Búsqueda	Cooling	7,2	10,9	84,5	86,1	1,7	29
XGBoost	Búsqueda	Cooling	7,3	9,3	80,7	85,9	5,1	36
RNN-lstm	Optimizado 2	Cooling	17	29,8	73	75	2	54
SVM	Búsqueda	Cooling	9,8	14,5	72,5	72,9	0,4	55
RNN-lstm	Optimizado	Cooling	19,8	32,6	68	67	1	60
RNN-lstm	Búsqueda	Cooling	20,6	36,1	60	69	9	62
Regresión Lineal	Búsqueda	Cooling	19,6	27,8	1,7	41,1	39,4	76

- En primer lugar, Random Forest presenta el mejor desempeño global en su configuración Optimizado 2 (R^2 Test = 91,1%), acompañado de errores relativamente bajos (MAE = 5; RMSE = 8,3). Esta mejora se asocia con un incremento en el número de estimadores (600) y la eliminación de la restricción en la profundidad máxima (max_depth = None), lo que permite capturar relaciones más complejas. Sin embargo, el aumento del Gap R (7,7) sugiere un nivel moderado de

sobreajuste, indicando que la ganancia en precisión viene acompañada de una ligera pérdida en capacidad de generalización.

Tabla 15. Configuraciones de hiperparámetros evaluadas en la variable Cooling.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	400	600	600	[400, 600]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	10	None	None	[None, 10]
Random Forest	min_samples_split	[2, 5, 7, 10]	5	5	2	[2, 5]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	1	2	1	[1, 2]
Random Forest	max_features	[log2, sqrt]	sqrt	log2	sqrt	[log2, sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	300	500	400	[300, 400, 500]
XGBoost	max_depth	[4, 6]	4	4	6	[4, 6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,005	0,005	0,03	[0.005, 0.03]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	0,8	1	1	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	1	1	5	[1, 5]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	50	10	100	[10, 50, 100]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,01	0,2	0,2	[0.01, 0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	scale	1	1	[scale, 1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,2	0,2	0,2	[0.2]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,001	0,003	0,003	[0.001, 0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(128,64,32)	(128,64,32)	(128,64,32)	[(128, 64,32)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Rmsprop	Rmsprop	Adam	[adam, rmsprop]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	168	336	168	[168, 336]
RNN-lstm	activation_function	[relu, tanh, elu]	Tanh	Tanh	Tanh	[tanh]
RNN-lstm	batch_size	[12, 32, 64, 128]	64	32	32	[32, 64]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- Por su parte, XGBoost muestra un comportamiento competitivo, especialmente en Optimizado 2 (R^2 Test = 87,9), con el menor MAE (4,9) entre los modelos evaluados. Este desempeño se relaciona con una tasa de aprendizaje más alta (0,03) y ajustes en colsample_bytree y min_child_weight, que permiten un aprendizaje más agresivo. No obstante, el Gap R (6,7) evidencia también cierto grado de sobreajuste, aunque menor que en Random Forest en términos relativos.

- El modelo SVM alcanza resultados intermedios, con su mejor configuración en Optimizado 2 (R^2 Test = 86,7), manteniendo un Gap R más controlado (3,4). Esto sugiere una mejor estabilidad del modelo frente a cambios en los datos, derivada del uso consistente del kernel RBF y valores moderados de C y gamma. Sin embargo, sus errores (MAE y RMSE) son superiores a los de los modelos basados en árboles, lo que limita su desempeño global.

- En contraste, las redes RNN-LSTM presentan un desempeño considerablemente inferior, con R^2 Test que no supera el 73% incluso en la mejor configuración. A pesar de ajustes en la arquitectura (capas (128,64,32)), tamaño de ventana y optimizador, los altos errores (RMSE > 29) indican dificultades para capturar adecuadamente la dinámica de la variable. Además, los valores de Gap R relativamente bajos en algunos casos no compensan la baja precisión predictiva.

- Finalmente, la Regresión Lineal muestra un desempeño muy deficiente (R^2 Test = 1,7), con un Gap R extremadamente alto (39,4), lo que evidencia una incapacidad del modelo para representar la complejidad no lineal de la variable Cooling.

4.5 Resultados de la variable Ventilation

Para la variable Ventilation, el análisis integrado de las métricas de error (MAE y RMSE), el coeficiente de determinación (R^2) y las configuraciones de hiperparámetros de la Tabla 16 y Tabla 17 respectivamente permite evaluar cómo las distintas etapas de optimización impactan tanto el desempeño predictivo como la estabilidad de los modelos.

Tabla 16. Desempeño de los modelos en la variable Ventilation.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
XGBoost	Optimizado 2	Ventilation	8,2	14,1	83,4	84,8	1,3	30
XGBoost	Optimizado	Ventilation	8,5	14,4	82,6	88,8	6,2	33
XGBoost	Búsqueda	Ventilation	9,7	15,2	80,7	77,2	3,5	35
Random Forest	Optimizado 2	Ventilation	7,5	13,3	80,4	91,8	11,4	37
SVM	Optimizado 2	Ventilation	7,9	13,9	78,4	82,7	4,3	42
SVM	Optimizado	Ventilation	8,4	14,4	76,9	79,3	2,4	45
RNN-lstm	Optimizado 2	Ventilation	10,3	17	76	73,7	2,3	47
Random Forest	Optimizado	Ventilation	9,3	14,8	75,7	77,9	2,2	48
RNN-lstm	Optimizado	Ventilation	11,2	17,3	74,9	70	4,9	50
SVM	Búsqueda	Ventilation	10,1	16	71,4	72,5	1	56
RNN-lstm	Búsqueda	Ventilation	13,3	19,4	68,7	67,3	1,4	59
Random Forest	Búsqueda	Ventilation	12,6	17,9	64,1	64,3	0,1	61
Regresión Lineal	Búsqueda	Ventilation	23,6	32,6	4,9	35	30,1	74

- En este caso, XGBoost se posiciona como el modelo con mejor desempeño global, alcanzando su mejor configuración en Optimizado 2 (R^2 Test = 83,4), acompañado de un bajo nivel de error (MAE = 8,2; RMSE = 14,1) y un Gap R reducido (1,3), lo que indica una adecuada capacidad de generalización. Este resultado se relaciona con una combinación balanceada de hiperparámetros,

particularmente una profundidad moderada (max_depth = 6), reducción en el número de estimadores (300) y una tasa de aprendizaje intermedia (0,03), lo que favorece un aprendizaje más controlado y menos propenso al sobreajuste en comparación con la fase Optimizado (Gap R = 6,2).

Tabla 17. Configuraciones de hiperparámetros evaluadas en la variable Ventilation.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	300	300	600	[300, 600]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	5	10	None	[None, 5, 10]
Random Forest	min_samples_split	[2, 5, 7, 10]	10	2	5	[2, 5, 10]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	4	1	2	[1, 2, 4]
Random Forest	max_features	[log2, sqrt]	sqrt	sqrt	sqrt	[sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	600	400	300	[300, 400, 600]
XGBoost	max_depth	[4, 6]	4	6	6	[4, 6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,005	0,05	0,03	[0.005, 0.03, 0.05]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	0,8	1	1	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	5	5	5	[5]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	100	10	50	[10, 50, 100]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,2	0,2	0,2	[0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	scale	1	1	[scale, 1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,2	0,5	0,5	[0.2, 0.5]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,003	0,001	0,003	[0.001, 0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(128,64,32)	(128,64,32)	(64,32,16)	[(128, 64,32), (64,32,16)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Adam	Rmsprop	Rmsprop	[adam, rmsprop]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	336	336	336	[336]
RNN-lstm	activation_function	[relu, tanh, elu]	Tanh	Relu	Relu	[relu, tanh]
RNN-lstm	batch_size	[12, 32, 64, 128]	64	32	32	[32, 64]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- Por su parte, Random Forest muestra un comportamiento menos estable. Aunque en Optimizado 2 logra un R^2 Test competitivo (80,4) y el menor RMSE (13,3), presenta un Gap R elevado (11,4), evidenciando un claro sobreajuste. Este comportamiento está asociado al incremento en el número de árboles (600) y la eliminación del límite de profundidad (max_depth = None), lo que incrementa la capacidad del modelo para ajustarse a los datos de entrenamiento, pero reduce su capacidad de generalización.

- El modelo SVM presenta un desempeño intermedio, con su mejor resultado en Optimizado 2 (R^2 Test = 78,4), manteniendo un equilibrio razonable entre error y estabilidad (Gap R = 4,3). La

consistencia en el uso del kernel RBF y valores moderados de C (50) y gamma (1) contribuyen a un modelo relativamente robusto, aunque con menor capacidad predictiva frente a XGBoost.

- En cuanto a las redes RNN-LSTM, los resultados son moderados, con un R^2 Test máximo de 76% en Optimizado 2. A pesar de ajustes en la arquitectura (reducción a (64,32,16)), tasa de aprendizaje y optimizador (RMSprop), el modelo no logra superar a los enfoques basados en árboles, lo que sugiere que la estructura temporal capturada no es suficiente para modelar eficazmente esta variable.

- Finalmente, la Regresión Lineal presenta un desempeño muy bajo (R^2 Test = 4,9) y un Gap R extremadamente alto (30,1), lo que confirma su incapacidad para capturar las relaciones no lineales presentes en la variable Ventilation.

4.6 Resultados de la variable Other electricity

En la variable Other electricity, la relación entre las métricas de desempeño (MAE, RMSE y R^2) y las configuraciones de hiperparámetros de la Tabla 18 y Tabla 19 respectivamente, permite evidenciar un comportamiento más inestable de los modelos, así como una mayor dificultad para capturar la dinámica subyacente de esta variable en comparación con otras analizadas.

Tabla 18. Desempeño de los modelos en la variable Other electricity.

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Other electricity	12,4	20,4	79,5	93,2	13,7	39
Random Forest	Optimizado	Other electricity	13,6	21,2	77,8	84,2	6,5	44
SVM	Optimizado 2	Other electricity	13,8	22,5	74,9	80,7	5,8	49
SVM	Optimizado	Other electricity	16,3	24,8	69,6	72,8	3,2	57
Random Forest	Búsqueda	Other electricity	17,5	25	69,1	72,3	3,1	58
XGBoost	Optimizado 2	Other electricity	15,2	23,8	57,4	82,2	24,8	64
XGBoost	Optimizado	Other electricity	15,2	24,1	56,5	83,8	27,3	65
SVM	Búsqueda	Other electricity	21,1	30,6	53,8	55,6	1,8	66
XGBoost	Búsqueda	Other electricity	15,4	24,9	53,6	91,1	37,5	67
RNN-lstm	Optimizado 2	Other electricity	24	33,9	13,8	61,1	47,3	73
RNN-lstm	Optimizado	Other electricity	27,2	35,8	3,7	28,5	24,8	75
RNN-lstm	Búsqueda	Other electricity	28,7	37,8	-6,9	20,5	27,4	77
Regresión Lineal	Búsqueda	Other electricity	42,6	47,3	-32,4	9,6	42	78

- En este contexto, Random Forest vuelve a posicionarse como el modelo con mejor desempeño, particularmente en su configuración Optimizado 2 (R^2 Test = 79,5), con errores relativamente contenidos (MAE = 12,4; RMSE = 20,4). Esta mejora se asocia con un aumento progresivo en el

número de estimadores (hasta 600) y la eliminación de la restricción en la profundidad del árbol (max_depth = None), lo que incrementa la capacidad del modelo para capturar relaciones complejas. Sin embargo, el Gap R elevado (13,7) indica un sobreajuste considerable, sugiriendo que parte de este desempeño no se generaliza adecuadamente a datos no vistos.

Tabla 19. Configuraciones de hiperparámetros evaluadas en la variable Other electricity.

Algoritmo	Hiperparametros	Espacio búsqueda	Busqueda	Optimizado	Optimizado 2	Espacio Optimizado
Random Forest	n_estimators	[100, 200, 300, 400, 500, 600]	300	400	600	[300, 400, 600]
Random Forest	max_depth	[None, 5, 10 , 15, 20]	10	20	None	[None, 10, 20]
Random Forest	min_samples_split	[2, 5, 7, 10]	2	10	5	[2, 5, 10]
Random Forest	min_samples_leaf	[1, 2, 4, 6]	1	6	1	[1, 6]
Random Forest	max_features	[log2, sqrt]	log2	log2	sqrt	[log2, sqrt]
Random Forest	max_samples	None	None	None	None	None
XGBoost	n_estimators	[200, 300, 400, 500, 600]	500	400	400	[400, 500]
XGBoost	max_depth	[4, 6]	6	4	4	[4, 6]
XGBoost	learning_rate	[0.003, 0.005, 0.03, 0.05]	0,05	0,05	0,03	[0.03, 0.05]
XGBoost	subsample	[0,8]	0,8	0,8	0,8	[0,8]
XGBoost	colsample_bytree	[0.8, 1]	1	1	0,8	[0.8, 1]
XGBoost	min_child_weight	[1, 5]	1	1	5	[1, 5]
XGBoost	gamma	0	0	0	0	0
XGBoost	reg_lambda	1	1	1	1	1
XGBoost	reg_alpha	0	0	0	0	0
SVM	kernel	[rbf, linear]	rbf	rbf	rbf	[rbf]
SVM	C	[1, 10, 50, 100, 200]	100	10	100	[10, 100]
SVM	epsilon	[0.01, 0.05, 0.1, 0.2]	0,2	0,2	0,2	[0.2]
SVM	gamma	[scale, auto, 0.01, 0.1, 1]	scale	1	1	[scale, 1]
RNN-lstm	dropout	[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]	0,2	0,2	0,5	[0.2, 0.5]
RNN-lstm	learning_rate	[0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005]	0,003	0,001	0,003	[0.001, 0.003]
RNN-lstm	num_layers + hidden_units	[(256,128,64), (128,64,32), (64,32,16),(32,16,8)]	(64,32,16)	(128,64,32)	(128,64,32)	[(128, 64,32), (64,32,16)]
RNN-lstm	optimizer	[adam, rmsprop, sgd, adagrad, adamax]	Rmsprop	Rmsprop	Rmsprop	[rmsprop]
RNN-lstm	window_size	[24, 48, 72, 168, 336]	336	168	168	[168, 336]
RNN-lstm	activation_function	[relu, tanh, elu]	Tanh	Tanh	Relu	[relu, tanh]
RNN-lstm	batch_size	[12, 32, 64, 128]	32	32	32	[32]
RNN-lstm	loss	mse	mse	mse	mse	mse
RNN-lstm	monitor	val_loss	val_loss	val_loss	val_loss	val_loss
RNN-lstm	patience	5	5	5	5	5
Regresión Lineal	-	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda	Sin busqueda

- El modelo SVM presenta un desempeño intermedio, alcanzando su mejor resultado en Optimizado 2 (R^2 Test = 74,9) con un Gap R más moderado (5,8). La estabilidad observada se relaciona con el uso consistente del kernel RBF y valores de hiperparámetros relativamente controlados (C = 100, epsilon = 0,2, gamma = 1), lo que favorece un balance entre ajuste y generalización, aunque con menor precisión que Random Forest.
- Por otro lado, XGBoost muestra un comportamiento menos favorable en esta variable, con R^2 Test que no supera el 57,4 incluso en su mejor configuración. A pesar de ajustes en parámetros como learning_rate (0,03–0,05), max_depth y colsample_bytree, el modelo presenta valores de Gap R extremadamente altos (hasta 37,5), lo que evidencia un fuerte sobreajuste. Esto sugiere que, para esta variable, el esquema de boosting no logra generalizar adecuadamente y tiende a memorizar los datos de entrenamiento.

- Las redes RNN-LSTM presentan un desempeño muy bajo, con valores de R^2 Test cercanos o incluso inferiores a cero en algunas configuraciones. A pesar de cambios en la arquitectura (capas (128,64,32)), funciones de activación y tamaño de ventana, los altos errores (RMSE > 33) y los elevados valores de Gap R indican una incapacidad significativa para modelar la variable, posiblemente debido a una baja señal temporal o alta variabilidad no estructurada.

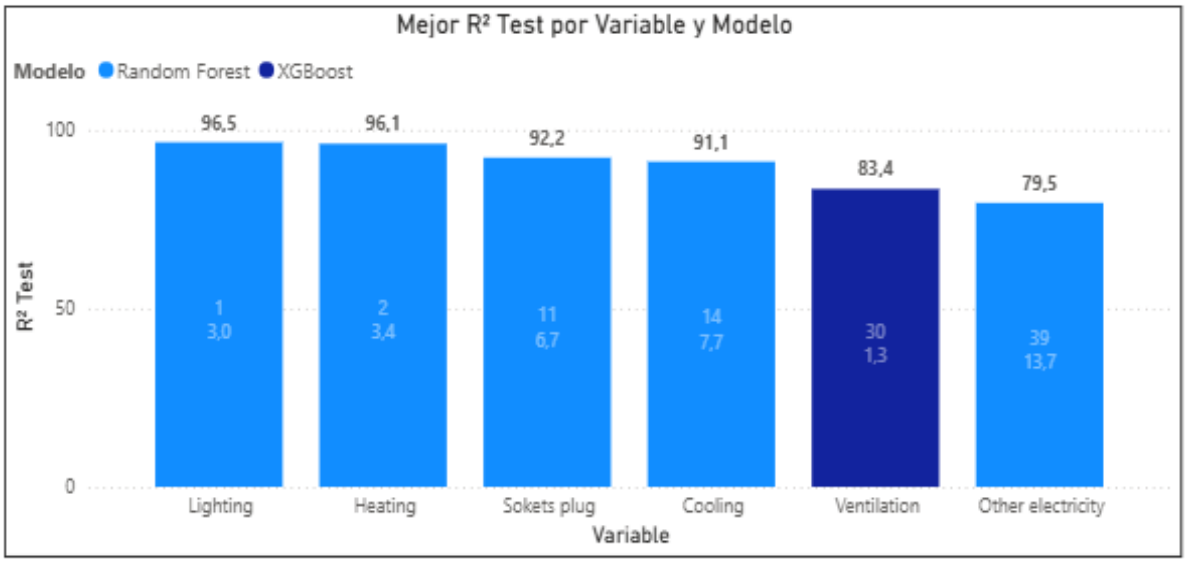
- Finalmente, la Regresión Lineal confirma su inadecuación para este problema, con un R^2 Test negativo (-32,4), lo que indica que el modelo es peor que una predicción basada en la media. El alto Gap R (42) refuerza la evidencia de un ajuste deficiente y una nula capacidad de generalización.

4.7 Mejores resultados de desempeño por variable

La Imagen 8 presenta los mejores resultados obtenidos para cada variable objetivo, considerando el modelo y la configuración de hiperparámetros que alcanzaron el mayor coeficiente de determinación (R^2) sobre el conjunto de prueba. Adicionalmente, se incluyen el Gap R^2, calculado como la diferencia entre el desempeño en entrenamiento y prueba, y la posición obtenida en el ranking global de modelos evaluados.

Los resultados muestran que Lighting obtuvo el mejor desempeño de todas las variables analizadas, alcanzando un R^2 Test de 96,5 % mediante el modelo Random Forest Optimizado 2. Este resultado estuvo acompañado de un Gap R^2 de 3,0 %, indicando una adecuada capacidad de generalización. De forma similar, la variable Heating alcanzó un R^2 Test de 96,1 % con un Gap R^2 de 3,4 %, lo que evidencia que los patrones de consumo asociados a calefacción presentan una estructura altamente predecible y pueden ser representados eficazmente mediante modelos basados en árboles de decisión.

Las variables Sockets plug y Cooling también presentaron resultados satisfactorios, con valores de R^2 Test de 92,2 % y 91,1 %, respectivamente. Sin embargo, ambas variables registraron diferencias más elevadas entre entrenamiento y prueba, reflejadas en Gap R^2 de 6,7 % y 7,7 %. Aunque estos valores indican una ligera tendencia al sobreajuste, los modelos mantienen una capacidad predictiva adecuada para aplicaciones de desagregación energética.



Modelo	Fase	Variable	MAE	RMSE	R² Test (%)	R² Train (%)	Gap R	Rank	Tiempo Ent (Seg)
Random Forest	Optimizado 2	Lighting	1,76	2,68	96,51	99,52	3,01	1	5,27
Random Forest	Optimizado 2	Heating	6,66	12,91	96,12	99,50	3,38	2	5,63
Random Forest	Optimizado 2	Sokets plug	1,37	1,93	92,20	98,87	6,67	11	4,7
Random Forest	Optimizado 2	Cooling	5,01	8,26	91,08	98,78	7,70	14	4,15
XGBoost	Optimizado 2	Ventilation	8,23	14,08	83,44	84,78	1,33	30	5,27
Random Forest	Optimizado 2	Other electricity	12,44	20,37	79,53	93,21	13,67	39	6,49

Imagen 8. Mejor R² por variable.

Para la variable Ventilation, el mejor resultado fue obtenido por XGBoost Optimizado 2, alcanzando un R² Test de 83,4 %. Aunque este desempeño es inferior al observado en las variables anteriores, el Gap R² de únicamente 1,3 % representa el menor valor de todas las variables analizadas. Esto demuestra una elevada estabilidad y capacidad de generalización, sugiriendo que el modelo mantiene un comportamiento consistente frente a datos no utilizados durante el entrenamiento.

La variable Other electricity presentó el mayor nivel de dificultad de modelado. El mejor resultado correspondió a Random Forest Optimizado 2 con un R² Test de 79,5 %, acompañado de un Gap R² de 13,7 %, el valor más alto entre todas las variables evaluadas. Este comportamiento puede atribuirse a la naturaleza heterogénea de esta categoría, que agrupa diferentes tipos de consumos eléctricos con patrones operativos diversos, dificultando la identificación de relaciones consistentes entre el consumo agregado y el consumo individual de la carga.

De manera global, los resultados evidencian que Random Forest Optimizado 2 fue el modelo con mejor desempeño para cinco de las seis variables analizadas, obteniendo los primeros lugares del ranking en Lighting y Heating, así como resultados superiores al 90 % de R² en Sockets plug y Cooling. Por su parte, XGBoost Optimizado 2 destacó únicamente en la variable Ventilation, donde logró el mejor equilibrio entre precisión y capacidad de generalización.

Estos resultados confirman que los modelos basados en ensambles de árboles constituyen la alternativa más efectiva para la desagregación no intrusiva de cargas en el conjunto de datos estudiado. Asimismo, evidencian que las variables asociadas a iluminación, calefacción y enchufes presentan patrones de consumo más estructurados y predecibles, mientras que categorías agregadas como Other electricity representan el principal desafío para los procesos de desagregación energética debido a su elevada variabilidad operativa.

Los resultados obtenidos demuestran que el proceso de entrenamiento, optimización y evaluación permitió identificar configuraciones de Machine Learning capaces de estimar con alta precisión el consumo energético desagregado de las diferentes cargas eléctricas, destacándose Random Forest Optimizado 2 como la alternativa de mejor desempeño global para el problema planteado.

4.8 Hiperparámetros óptimos identificados

Como resultado del proceso de optimización descrito en el Capítulo 3, se identificaron configuraciones específicas de hiperparámetros para cada modelo y variable objetivo. La Tabla 20 presenta los hiperparámetros asociados a los modelos con mejor desempeño en el conjunto de prueba.

Los resultados muestran que el algoritmo Random Forest fue seleccionado como el mejor modelo para cinco de las seis variables analizadas (Lighting, Heating, Sockets plug, Cooling y Other electricity). En estos casos se observa una alta consistencia en la configuración óptima obtenida, destacándose el uso recurrente de profundidades máximas sin restricción (max_depth = None), divisiones mínimas de dos muestras (min_samples_split = 2) y una sola muestra por hoja (min_samples_leaf = 1). Asimismo, la opción sqrt fue seleccionada en todos los casos para el parámetro max_features, indicando que el uso de una fracción aleatoria de variables contribuye positivamente a la capacidad de generalización del modelo.

Tabla 20. Hiperparámetros asociados a los modelos con mejor desempeño

Variable	Modelo ganador	n_estimators	max_ depth	min_ samples_ split	min_ samples _	max_ features	learning_ rate	subsample	colsample_ bytree	min_ child_ weight	gamma	reg_ lambd a	reg_ alpha	R^2 Test (%)
Lighting	Random Forest	600	None	5	1	sqrt	-	-	-	-	-	-	-	96.5
Heating	Random Forest	600	None	2	1	sqrt	-	-	-	-	-	-	-	96.1
Sockets plug	Random Forest	300	None	2	1	sqrt	-	-	-	-	-	-	-	92.2
Cooling	Random Forest	600	None	2	1	sqrt	-	-	-	-	-	-	-	91.1
Ventilation	XGBoost	300	6	-	-	-	0,03	0,8	1	5	0	1	0	83.4
Other electricity	Random Forest	600	None	2	1	sqrt	-	-	-	-	-	-	-	79.5

En relación con el número de árboles (n_estimators), se identificaron dos configuraciones predominantes. Para las variables Ventilation y Sockets plug se obtuvieron mejores resultados con 300 árboles, mientras que Lighting, Other electricity, Cooling y Heating alcanzaron un desempeño superior con 600 árboles. Este

comportamiento sugiere que el incremento en el número de estimadores favorece la estabilidad del modelo en variables con patrones de consumo más complejos.

Por otra parte, la variable Ventilation presentó un comportamiento diferenciado, siendo XGBoost el modelo con mejor desempeño. La configuración óptima estuvo compuesta por 300 estimadores, profundidad máxima de seis niveles y una tasa de aprendizaje de 0,03. Adicionalmente, los parámetros de regularización (reg_lambda = 1 y reg_alpha = 0) permitieron mantener un adecuado equilibrio entre capacidad predictiva y generalización, reflejado en el bajo valor de Gap R^2 obtenido para esta variable.

Los resultados obtenidos evidencian que los modelos basados en árboles de decisión, particularmente Random Forest, presentan una alta capacidad para representar los patrones de consumo energético presentes en el conjunto de datos analizado. Asimismo, la estabilidad observada en los hiperparámetros óptimos seleccionados constituye un indicador de la robustez alcanzada durante el proceso de optimización.

4.9 Reconciliación de modelos

Debido a que la estrategia propuesta utiliza modelos independientes para estimar cada una de las cargas energéticas (Ventilation, Sockets plug, Lighting, Other electricity, Cooling y Heating), se identificó una situación frecuente en los problemas de desagregación no intrusiva de cargas (NILM): la suma de las predicciones individuales no coincide exactamente con el consumo energético total observado.

Esta inconsistencia surge porque cada modelo es entrenado y optimizado de manera independiente, sin incorporar restricciones que garanticen la conservación de energía entre las variables predichas. Como consecuencia, aunque los modelos puedan presentar un alto desempeño individual, la suma de sus estimaciones puede diferir del consumo agregado registrado por el medidor principal.

Con el fin de garantizar coherencia energética, se implementó un procedimiento de reconciliación proporcional. Este método ajusta las predicciones individuales conservando la participación relativa estimada por cada modelo y asegurando que la suma final de los consumos desagregados sea exactamente igual al consumo total observado.

La aplicación de este procedimiento permitió eliminar las inconsistencias existentes entre las predicciones individuales y el consumo agregado, garantizando el cumplimiento del principio de conservación de energía requerido en aplicaciones de monitoreo y gestión energética

Los resultados obtenidos evidencian que la reconciliación constituye una etapa necesaria dentro del proceso de desagregación energética, ya que permite integrar las predicciones generadas por los diferentes modelos en una solución coherente desde el punto de vista físico y operativo. De esta manera, las estimaciones finales pueden ser utilizadas para análisis energéticos posteriores sin presentar discrepancias entre el consumo total y la suma de sus componentes.

Dado que los modelos fueron entrenados de forma independiente para cada categoría de carga, las predicciones individuales no garantizan por construcción que su suma coincida con el consumo total observado. Para corregir esta inconsistencia se implementó un procedimiento de reconciliación proporcional, cuyo fundamento teórico se describe en la Sección 1.3.2 del marco teórico. En este procedimiento cada predicción individual y_i se ajusta mediante el factor $y_{total} / \hat{y}_{total}$, donde y_{total} es el consumo total real observado y $\hat{y}_{total} = \Sigma y_i$ es la suma de las predicciones base, obteniendo el valor

reconciliado $yR_i = (y_i / \hat{y}_{total}) \times y_{total}$. Este ajuste garantiza que $\Sigma\, yR_i = y_{total}$, cumpliendo el principio de conservación de energía sin alterar las proporciones relativas estimadas por los modelos base. Los resultados de su aplicación se presentan en la Sección 4.9.

4.10 Evaluación estadística resultados del error en la reconciliación.

El análisis estadístico del error de reconciliación evidenció una distribución asimétrica con predominio de valores negativos, indicando una tendencia general de los modelos base a subestimar el consumo total. La media del error fue de -38.09 y la mediana de -26.24, confirmando que la mayor parte de las discrepancias se concentran en valores inferiores a cero.

El histograma mostró que los errores se distribuyen principalmente entre -60 y 0, aunque se identificaron valores extremos tanto positivos como negativos. El boxplot evidenció una alta dispersión de los errores, con un rango intercuartílico de aproximadamente 48.92, además de múltiples valores atípicos asociados a posibles periodos de alta demanda energética o variaciones abruptas en el consumo. Este análisis puede observarse en las Imagen 9 e Imagen 10 Imagen 10.
Los valores extremos alcanzaron mínimos de -259 y máximos de 263.8, indicando la existencia de inconsistencias significativas en determinados instantes temporales. Asimismo, las métricas de error obtenidas ($MAE = 44.13$, $RMSE = 62.90$ y desviación estándar $= 50.06$) reflejan una variabilidad considerable entre las predicciones individuales y el consumo total observado.
En conjunto, estos resultados demuestran que, aunque los modelos base logran capturar parcialmente la dinámica del consumo energético, persisten discrepancias importantes entre las predicciones desagregadas y el consumo agregado. Por tanto, la aplicación de técnicas de reconciliación resulta necesaria para garantizar coherencia energética y reducir las inconsistencias presentes en las predicciones originales.

Los resultados obtenidos evidencian que los modelos de aprendizaje basados en ensambles alcanzan el mejor desempeño para la desagregación del consumo energético, destacándose Random Forest como el algoritmo con mayor capacidad predictiva en la mayoría de las categorías evaluadas. La optimización de hiperparámetros permitió mejorar de forma consistente las métricas de desempeño, mientras que el proceso de reconciliación garantizó la coherencia entre la suma de los consumos desagregados y la medición agregada. En conjunto, estos resultados demuestran el cumplimiento de los objetivos específicos relacionados con la evaluación de modelos, la reconciliación de predicciones y la validación cuantitativa del sistema propuesto.

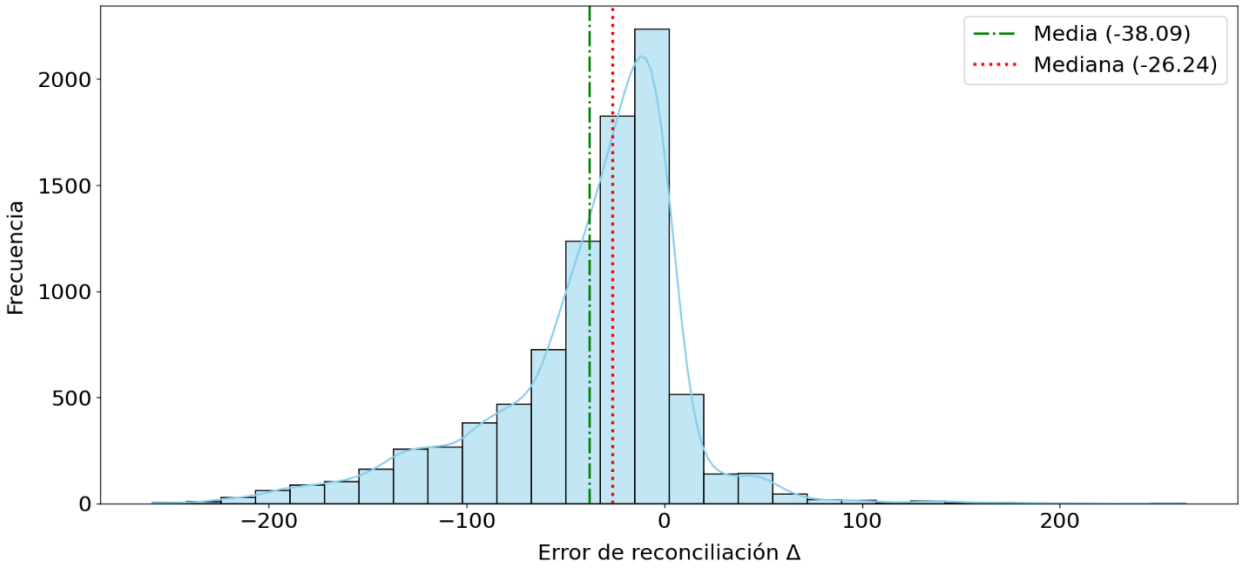


Imagen 9. Distribución del error de la reconciliación.

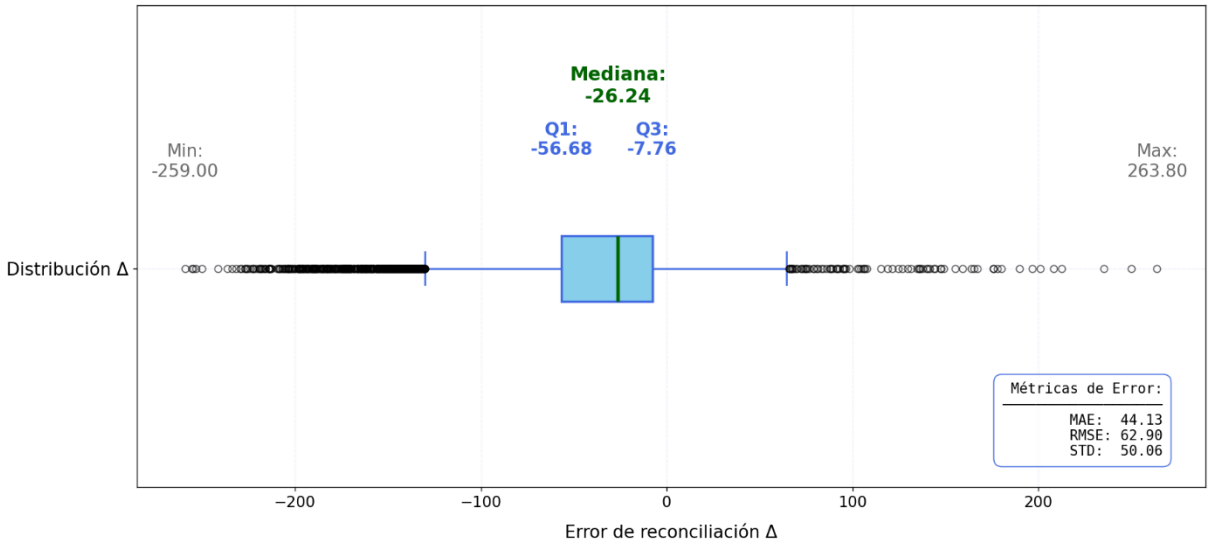


Imagen 10. Boxplot del error de la reconciliación.

5. CONCLUSIONES

La investigación permitió desarrollar y evaluar un modelo de desagregación energética basado en técnicas de aprendizaje automático, demostrando la viabilidad del enfoque NILM para estimar el consumo de diferentes categorías de carga en edificios comerciales. Los resultados obtenidos evidencian que el objetivo general fue alcanzado, al obtener modelos con altos niveles de precisión para varias de las cargas analizadas y establecer un procedimiento reproducible de entrenamiento, optimización y reconciliación de predicciones.

El primer objetivo específico se cumplió mediante el procesamiento y análisis exploratorio del conjunto de datos, identificando patrones de consumo, valores atípicos e inconsistencias temporales que fueron tratados para garantizar la calidad de la información utilizada durante el entrenamiento de los modelos.

El segundo objetivo específico se alcanzó mediante la implementación y evaluación comparativa de diferentes algoritmos de aprendizaje automático, evidenciando que los modelos basados en ensambles, especialmente Random Forest y XGBoost, presentaron el mejor desempeño para la mayoría de las categorías de carga analizadas.

El tercer objetivo específico se cumplió mediante la implementación del mecanismo de reconciliación proporcional, el cual permitió garantizar la coherencia entre las predicciones individuales y el consumo energético agregado, reduciendo inconsistencias derivadas del entrenamiento independiente de los modelos.

Finalmente, el cuarto objetivo específico se logró mediante la evaluación cuantitativa del desempeño utilizando métricas como MAE, RMSE, R^2 y Gap R, lo que permitió seleccionar objetivamente los modelos con mejor capacidad predictiva y analizar su comportamiento frente a diferentes categorías de consumo.

Los altos valores de R^2 alcanzados en variables como Lighting, Heating, Sockets plug y Cooling (superiores al 90% en varios modelos) evidencian que el conjunto de datos contiene patrones suficientemente claros y distinguibles para ciertos tipos de consumo. Esto sugiere que, al menos para estas cargas, sí es viable realizar desagregación energética con un alto nivel de precisión, lo cual es un indicio positivo para la aplicación de NILM.

En contraste, variables como Other electricity y, en menor medida, Ventilation, presentan desempeños considerablemente más bajos y mayores valores de Gap R. Esto indica que dichas cargas son más heterogéneas, ruidosas o menos representativas en los datos, lo que dificulta su modelado. En el contexto de NILM, esto es consistente con la literatura: las cargas agregadas o “residuales” suelen ser más difíciles de descomponer, lo que implica que la desagregación completa del consumo total podría tener limitaciones importantes en estas categorías.

Adicionalmente, el hecho de que modelos como Random Forest y XGBoost logren capturar gran parte de la variabilidad en varias variables sugiere que existen relaciones no lineales aprovechables entre el consumo agregado y los consumos individuales, lo cual es un requisito clave para enfoques NILM basados en aprendizaje supervisado. Sin embargo, los incrementos en el Gap R en configuraciones más complejas evidencian riesgos de sobreajuste, lo que implica que la generalización del modelo a nuevos datos debe

ser cuidadosamente validada.

Por otro lado, el bajo desempeño de modelos como RNN-LSTM sugiere que, en este conjunto de datos, la dependencia temporal no es suficientemente fuerte o no está adecuadamente representada para justificar el uso de modelos secuenciales complejos. Esto puede indicar que el NILM en este caso podría beneficiarse más de enfoques basados en características agregadas o modelos de ensamble que de arquitecturas profundas temporales.

En conjunto, los resultados permiten concluir que sí es factible aplicar NILM en el conjunto de datos, pero de manera parcial y dependiente del tipo de carga. Las cargas bien definidas y con patrones consistentes pueden ser desagregadas con alta precisión, mientras que las cargas más difusas requerirán estrategias adicionales, como una mejor ingeniería de variables, segmentación de cargas o modelos híbridos.

Finalmente, el proceso de reconciliación proporcional demostró ser una estrategia efectiva para garantizar coherencia energética entre las predicciones individuales y el consumo total observado. La aplicación de este método permitió corregir las inconsistencias generadas por los modelos independientes, asegurando que la suma de los consumos desagregados coincidiera exactamente con el consumo total del sistema

Más allá del desempeño alcanzado por los modelos evaluados, esta investigación aporta una metodología reproducible para el desarrollo de sistemas NILM basada en el procesamiento sistemático de datos, la optimización de hiperparámetros, la evaluación comparativa de algoritmos y la reconciliación de predicciones. Este enfoque puede servir como referencia para futuras investigaciones orientadas a mejorar la eficiencia energética en edificios comerciales y apoyar la toma de decisiones basada en datos.

6. TRABAJOS FUTUROS

A partir de los resultados obtenidos y considerando la viabilidad parcial del enfoque NILM en este conjunto de datos, se identifican varias líneas de trabajo futuro orientadas a mejorar el desempeño y la aplicabilidad práctica del modelo:

En primer lugar, se recomienda profundizar en la ingeniería de características (feature engineering). La incorporación de variables derivadas, como rezagos temporales, variables climáticas (temperatura, humedad) o indicadores de comportamiento (horarios, días de la semana), podría mejorar significativamente la capacidad de los modelos para capturar patrones más complejos, especialmente en variables con bajo desempeño como Other electricity y Ventilation.

En segundo lugar, es pertinente explorar enfoques híbridos de modelado, combinando modelos de ensamble (como Random Forest o XGBoost) con técnicas especializadas en NILM, tales como modelos probabilísticos o arquitecturas basadas en descomposición de señales. Esto permitiría aprovechar tanto la capacidad predictiva como la interpretabilidad del proceso de desagregación.

Otra línea relevante consiste en el uso de modelos jerárquicos o por etapas, donde primero se identifiquen las cargas más representativas (por ejemplo, Lighting y Heating) y posteriormente se modele el residuo (Other electricity). Este enfoque podría mejorar la precisión global al reducir la complejidad del problema en cada etapa.

Finalmente, se plantea como trabajo futuro la optimización multiobjetivo, donde no solo se maximice el R^2, sino que también se minimice el Gap R, buscando un equilibrio entre precisión y generalización. Esto es especialmente relevante para evitar sobreajuste y garantizar la aplicabilidad práctica del modelo en entornos reales.

En conjunto, estas líneas de trabajo permitirían avanzar hacia sistemas NILM más robustos, precisos y escalables, capaces de abordar la complejidad inherente a la desagregación del consumo energético en contextos reales.

7. REFERENCIAS BIBLIOGRÁFICAS

[1] M. P. Castañeda, A. Reyes, y M. Cardozo, «Sector eléctrico colombiano: retos y oportunidades», 1 de mayo de 2021.

[2] «Trends and Projections in Europe 2023». Accedido: 26 de mayo de 2025. [En línea]. Disponible en: https://www.eea.europa.eu/en/analysis/publications/trends-and-projections-in-europe-2023

[3] G. W. Hart, «Nonintrusive appliance load monitoring», *Proceedings of the IEEE*, vol. 80, n.º 12, pp. 1870-1891, dic. 1992, doi: 10.1109/5.192069.

[4] «Energy Efficiency 2022 – Analysis», IEA. Accedido: 22 de mayo de 2025. [En línea]. Disponible en: https://www.iea.org/reports/energy-efficiency-2022

[5] J. Z. Kolter y T. Jaakkola, «Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation», en *Artificial Intelligence and Statistics*, PMLR, mar. 2012, pp. 1472-1482. Accedido: 22 de mayo de 2025. [En línea]. Disponible en: https://proceedings.mlr.press/v22/zico12.html

[6] M. I. Azad, R. Rajabi, y A. Estebsari, «Sequence-to-Sequence Model with Transformer-based Attention Mechanism and Temporal Pooling for Non-Intrusive Load Monitoring», 8 de junio de 2023, *arXiv*: arXiv:2306.05012. doi: 10.48550/arXiv.2306.05012.

[7] R. Sun, K. Dong, y J. Zhao, «DiffNILM: A Novel Framework for Non-Intrusive Load Monitoring Based on the Conditional Diffusion Model», *Sensors*, vol. 23, n.º 7, Art. n.º 7, ene. 2023, doi: 10.3390/s23073540.

[8] J. Kelly y W. Knottenbelt, «The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes», *Sci Data*, vol. 2, n.º 1, p. 150007, mar. 2015, doi: 10.1038/sdata.2015.7.

[9] C. Zhang, M. Zhong, Z. Wang, N. Goddard, y C. Sutton, «Sequence-to-point learning with neural networks for nonintrusive load monitoring», 18 de septiembre de 2017, *arXiv*: arXiv:1612.09106. doi: 10.48550/arXiv.1612.09106.

[10] A. Faustine, N. H. Mvungi, S. Kaijage, y K. Michael, «A Survey on Non-Intrusive Load Monitoring Methods and Techniques for Energy Disaggregation Problem», arXiv.org. Accedido: 22 de mayo de 2025. [En línea]. Disponible en: https://arxiv.org/abs/1703.00785v3

[11] R. Bonfigli, E. Principi, M. Fagiani, M. Severini, S. Squartini, y F. Piazza, «Non-intrusive load monitoring by using active and reactive power in additive Factorial Hidden Markov Models», *Applied Energy*, vol. 208, pp. 1590-1607, dic. 2017, doi: 10.1016/j.apenergy.2017.08.203.

[12] J. Kelly y W. Knottenbelt, «Neural NILM: Deep Neural Networks Applied to Energy Disaggregation», en *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, en BuildSys '15. New York, NY, USA: Association for Computing Machinery, nov. 2015, pp. 55-64. doi: 10.1145/2821650.2821672.

[13] M. Kaselimi, E. Protopapadakis, A. Voulodimos, N. Doulamis, y A. Doulamis, «Towards trustworthy Energy Disaggregation: A review of challenges, methods and perspectives for Non-

Intrusive Load Monitoring», 5 de julio de 2022, *arXiv*: arXiv:2207.02009. doi: 10.48550/arXiv.2207.02009.
[14] X. Zhou, S. Li, C. Liu, H. Zhu, N. Dong, y T. Xiao, «Non-Intrusive Load Monitoring Using a CNN-LSTM-RF Model Considering Label Correlation and Class-Imbalance», *IEEE Access*, vol. 9, pp. 84306-84315, 2021, doi: 10.1109/ACCESS.2021.3087696.
[15] T. Chen y C. Guestrin, «XGBoost: A Scalable Tree Boosting System», en *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ago. 2016, pp. 785-794. doi: 10.1145/2939672.2939785.
[16] V. Piccialli y A. M. Sudoso, «Improving Non-Intrusive Load Disaggregation through an Attention-Based Deep Neural Network», *Energies*, vol. 14, n.º 4, p. 847, feb. 2021, doi: 10.3390/en14040847.
[17] M. M. R. Khan, M. A. B. Siddique, y S. Sakib, «Non-Intrusive Electrical Appliances Monitoring and Classification using K-Nearest Neighbors», en *2019 2nd International Conference on Innovation in Engineering and Technology (ICIET)*, dic. 2019, pp. 1-5. doi: 10.1109/ICIET48527.2019.9290671.
[18] L. de Diego-Otón, D. Fuentes-Jimenez, Á. Hernández, y R. Nieto, «Recurrent LSTM Architecture for Appliance Identification in Non-Intrusive Load Monitoring», en *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, may 2021, pp. 1-6. doi: 10.1109/I2MTC50364.2021.9460046.
[19] N. Rajkumar, A. Govindaram, R. Geetha, S. A S F, N. S., y J. A. A., «Non-Intrusive Load Monitoring Techniques for Intelligent Energy Management: A Comparative Study of FHMM and LSTM Approach», en *2025 International Conference on Visual Analytics and Data Visualization (ICVADV)*, mar. 2025, pp. 869-874. doi: 10.1109/ICVADV63329.2025.10961750.
[20] S. Naderian, «A Novel Hybrid Deep Learning Approach for Non-Intrusive Load Monitoring of Residential Appliance Based on Long Short Term Memory and Convolutional Neural Networks», 15 de abril de 2021, *arXiv*: arXiv:2104.07809. doi: 10.48550/arXiv.2104.07809.
[21] Z. Xiao, W. Gang, J. Yuan, Y. Zhang, y C. Fan, «Cooling load disaggregation using a NILM method based on random forest for smart buildings», *Sustainable Cities and Society*, vol. 74, p. 103202, nov. 2021, doi: 10.1016/j.scs.2021.103202.

8. ANEXOS.

Repositorio de código fuente

Con el fin de garantizar la reproducibilidad de los resultados presentados en este documento, el código fuente desarrollado para el procesamiento de datos, el entrenamiento y optimización de los modelos de machine learning, y el proceso de reconciliación se encuentra disponible públicamente en el siguiente repositorio: https://github.com/Jotajaimes/proyecto_grado. El repositorio incluye los notebooks de análisis exploratorio y limpieza de datos, los scripts de entrenamiento y optimización de hiperparámetros para cada modelo evaluado (Regresión Lineal, SVM, Random Forest, XGBoost y RNN-LSTM), así como el script correspondiente al mecanismo de reconciliación proporcional descrito en el Capítulo 4.

Tabla 21. Resultados de desempeño de modelos de regresión

Modelo	Fase	Variable	MAE	RMSE	R^2 Test (%)	R^2 Train (%)	Gap R	Rank
Random Forest	Optimizado 2	Lighting	1,8	2,7	96,5	99,5	3	1
Random Forest	Optimizado 2	Heating	6,7	12,9	96,1	99,5	3,4	2
SVM	Optimizado 2	Lighting	1,9	3	95,7	97,6	2	3
SVM	Optimizado 2	Heating	6,8	14	95,5	97	1,5	4
Random Forest	Optimizado	Lighting	2	3,1	95,4	97,4	2,1	5
XGBoost	Optimizado 2	Heating	8,8	15,7	95,1	99,2	4	6
Random Forest	Optimizado	Heating	9,3	16,3	93,8	95,8	2	7
XGBoost	Optimizado 2	Lighting	1,8	3,1	93,7	98,6	4,9	8
SVM	Optimizado	Lighting	2,3	3,6	93,6	95,2	1,6	9
XGBoost	Optimizado	Heating	12	19,3	92,6	95,6	3	10
Random Forest	Optimizado 2	Sockets plug	1,4	1,9	92,2	98,9	6,7	11
SVM	Optimizado	Heating	10,2	18,6	92	93,2	1,2	12
Random Forest	Búsqueda	Heating	11,9	19	91,6	93,1	1,4	13
Random Forest	Optimizado 2	Cooling	5	8,3	91,1	98,8	7,7	14
Random Forest	Optimizado	Sockets plug	1,5	2,1	90,8	95,7	4,8	15
Random Forest	Búsqueda	Lighting	2,9	4,4	90,6	91,4	0,7	16
Random Forest	Optimizado	Cooling	5,2	8,5	90,6	96,1	5,5	17
SVM	Optimizado 2	Sockets plug	1,5	2,2	89,6	94,4	4,8	18
XGBoost	Optimizado	Lighting	3,1	4,1	89,1	92,2	3,1	19
XGBoost	Búsqueda	Heating	17,7	24	88,6	91,9	3,3	20
XGBoost	Optimizado 2	Cooling	4,9	7,4	87,9	94,6	6,7	21
SVM	Búsqueda	Lighting	3,3	5,1	87,2	88	0,8	22
SVM	Optimizado 2	Cooling	6,1	10,1	86,7	90,1	3,4	23
XGBoost	Optimizado	Cooling	5,6	7,8	86,4	90,5	4,1	24
SVM	Optimizado	Sockets plug	1,7	2,6	86,2	89,7	3,5	25
SVM	Búsqueda	Heating	14,1	24,7	85,8	86,5	0,6	26

SVM	Optimizado	Cooling	6,4	10,5	85,7	88,7	3	27
XGBoost	Búsqueda	Lighting	3,7	4,8	85,4	88,1	2,7	28
Random Forest	Búsqueda	Cooling	7,2	10,9	84,5	86,1	1,7	29
XGBoost	Optimizado 2	Ventilation	8,2	14,1	83,4	84,8	1,3	30
RNN-lstm	Optimizado 2	Heating	25,1	46,6	83	95	12	31
Random Forest	Búsqueda	Sockets plug	2	2,9	82,7	84,5	1,8	32
XGBoost	Optimizado	Ventilation	8,5	14,4	82,6	88,8	6,2	33
RNN-lstm	Optimizado	Heating	26,4	47,7	82	93	11	34
XGBoost	Búsqueda	Ventilation	9,7	15,2	80,7	77,2	3,5	35
XGBoost	Búsqueda	Cooling	7,3	9,3	80,7	85,9	5,1	36
Random Forest	Optimizado 2	Ventilation	7,5	13,3	80,4	91,8	11,4	37
SVM	Búsqueda	Sockets plug	2	3,1	79,6	82	2,3	38
Random Forest	Optimizado 2	Other electricity	12,4	20,4	79,5	93,2	13,7	39
XGBoost	Optimizado 2	Sockets plug	2	3,1	79,2	95,3	16,1	40
RNN-lstm	Búsqueda	Heating	27,9	51,3	79	93	14	41
SVM	Optimizado 2	Ventilation	7,9	13,9	78,4	82,7	4,3	42
RNN-lstm	Optimizado 2	Lighting	3,9	5,9	78	86,1	8,1	43
Random Forest	Optimizado	Other electricity	13,6	21,2	77,8	84,2	6,5	44
SVM	Optimizado	Ventilation	8,4	14,4	76,9	79,3	2,4	45
XGBoost	Optimizado	Sockets plug	2,1	3,3	76,9	95,4	18,5	46
RNN-lstm	Optimizado 2	Ventilation	10,3	17	76	73,7	2,3	47
Random Forest	Optimizado	Ventilation	9,3	14,8	75,7	77,9	2,2	48
SVM	Optimizado 2	Other electricity	13,8	22,5	74,9	80,7	5,8	49
RNN-lstm	Optimizado	Ventilation	11,2	17,3	74,9	70	4,9	50
XGBoost	Búsqueda	Sockets plug	2,3	3,4	74,6	90,9	16,2	51
RNN-lstm	Optimizado	Lighting	4,2	6,4	73,6	92,9	19,3	52
RNN-lstm	Búsqueda	Lighting	4,4	6,4	73,5	92,4	18,9	53
RNN-lstm	Optimizado 2	Cooling	17	29,8	73	75	2	54
SVM	Búsqueda	Cooling	9,8	14,5	72,5	72,9	0,4	55
SVM	Búsqueda	Ventilation	10,1	16	71,4	72,5	1	56
SVM	Optimizado	Other electricity	16,3	24,8	69,6	72,8	3,2	57
Random Forest	Búsqueda	Other electricity	17,5	25	69,1	72,3	3,1	58
RNN-lstm	Búsqueda	Ventilation	13,3	19,4	68,7	67,3	1,4	59
RNN-lstm	Optimizado	Cooling	19,8	32,6	68	67	1	60
Random Forest	Búsqueda	Ventilation	12,6	17,9	64,1	64,3	0,1	61
RNN-lstm	Búsqueda	Cooling	20,6	36,1	60	69	9	62
Regresión Lineal	Búsqueda	Heating	33,2	44,5	57,7	59,6	1,9	63
XGBoost	Optimizado 2	Other electricity	15,2	23,8	57,4	82,2	24,8	64
XGBoost	Optimizado	Other electricity	15,2	24,1	56,5	83,8	27,3	65
SVM	Búsqueda	Other electricity	21,1	30,6	53,8	55,6	1,8	66
XGBoost	Búsqueda	Other electricity	15,4	24,9	53,6	91,1	37,5	67

RNN-lstm	Búsqueda	Sockets plug	3,3	4,7	51	81,5	30,5	68
RNN-lstm	Optimizado 2	Sockets plug	3,4	4,9	48,3	58,6	10,4	69
RNN-lstm	Optimizado	Sockets plug	3,4	2	45,2	78,5	33,3	70
Regresión Lineal	Búsqueda	Lighting	6,6	9,6	35,1	33,7	1,5	71
Regresión Lineal	Búsqueda	Sockets plug	3,9	5,3	28,1	27,8	0,3	72
RNN-lstm	Optimizado 2	Other electricity	24	33,9	13,8	61,1	47,3	73
Regresión Lineal	Búsqueda	Ventilation	23,6	32,6	4,9	35	30,1	74
RNN-lstm	Optimizado	Other electricity	27,2	35,8	3,7	28,5	24,8	75
Regresión Lineal	Búsqueda	Cooling	19,6	27,8	1,7	41,1	39,4	76
RNN-lstm	Búsqueda	Other electricity	28,7	37,8	-6,9	20,5	27,4	77
Regresión Lineal	Búsqueda	Other electricity	42,6	47,3	-32,4	9,6	42	78

¡Error! Vínculo no válido.