



GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

TRABAJO DE FIN DE GRADO

**MODELOS ESTADÍSTICOS PARA LA PREDICCIÓN DE LA  
DEMANDA ELÉCTRICA**

Autor: Isabel Katalin Garate Aspichueta

Director: Pablo Carlos del Saz-Orozco Huang

Madrid

Junio 2025

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título  
Modelos estadísticos para la predicción de la demanda eléctrica  
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el  
curso académico 2024/25 es de mi autoría, original e inédito y  
no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha  
sido tomada de otros documentos está debidamente referenciada.



Fdo.: Isabel Katalin Garate Aspichueta

Fecha: 24/ 06/ 2025

Autorizada la entrega del proyecto  
EL DIRECTOR DEL PROYECTO

Fdo.: Pablo del Saz-Orozco Huang

Fecha: 23/06/2025





GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

TRABAJO DE FIN DE GRADO

**MODELOS ESTADÍSTICOS PARA LA PREDICCIÓN DE LA  
DEMANDA ELÉCTRICA**

Autor: Isabel Katalin Garate Aspichueta

Director: Pablo Carlos del Saz-Orozco Huang

Madrid

Junio 2025

# MODELOS ESTADÍSTICOS PARA LA PREDICCIÓN DE LA DEMANDA ELÉCTRICA

**Autor: Garate Aspichueta, Isabel Katalin.**

Director: del Saz-Orozco Huang, Pablo Carlos.

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

## RESUMEN DEL PROYECTO

### 1. Introducción

Este proyecto se basa en el desarrollo de modelos estadísticos avanzados (ARIMA, SARIMA y XGBoost) para predecir la demanda eléctrica en España, integrando técnicas de explicabilidad (SHAP y LIME) a estos modelos para entender los factores que influyen en las predicciones.

### 2. Definición del proyecto

El objetivo principal del proyecto es el desarrollo de modelos estadísticos predictivos avanzados y la aplicación de técnicas de explicabilidad para construir un sistema predictivo que permita anticipar el consumo eléctrico en la península.

Para ello, primero se ha realizado un análisis profundo de la tendencia de la demanda en los últimos años y de los factores que afectan al consumo. También se ha investigado en los modelos predictivos avanzados y las técnicas de explicabilidad.

Después, se han recopilado y procesado los datos desde 2019 hasta 2022 de la demanda eléctrica y de aquellas variables que se ha considerado que influyen en el consumo: temperatura, tipo de día y festividad.

A continuación, se han entrenado tres modelos ARIMA, SARIMA y XGBoost, analizándolos con métricas comparativas como RMSE y MAE.

Finalmente, se han analizado estos modelos aplicando las técnicas de explicabilidad SHAP y LIME, además de análisis cualitativos.

Finalmente se han extraído conclusiones acerca de las variables más influyentes en la demanda y sobre cómo mejorar los resultados de los modelos.

### 3. Descripción del modelo/sistema/herramienta

El proyecto ha sido desarrollado en Python, utilizando Anaconda en el entorno de Jupyter Notebook. Se han empleado librerías como pandas, pmdarima, XGBoost, o SHAP. El pipeline incluye la extracción de datos de Red Eléctrica de España, el preprocesamiento de la serie temporal horaria, la extracción y pre-procesamiento de variables (incluyendo temperatura y festivos), y la evaluación comparada de los modelos. XGBoost ha sido el más preciso, destacando también por su explicabilidad. La técnica SHAP ha proporcionado un entendimiento más global de la manera en la que las variables influyen en las predicciones, mientras que LIME ofrece

explicaciones más locales en un momento concreto. Ambas técnicas son combinables y sus resultados están alineados.

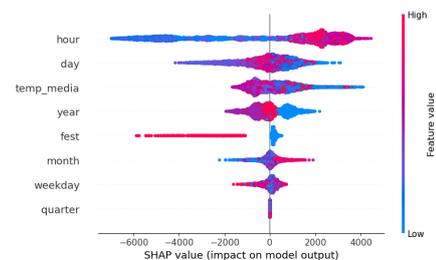
#### 4. Resultados

El modelo SARIMA ofrece mejores resultados que el modelo ARIMA por incorporar el término  $s$ , de estacionalidad en una serie temporal que claramente presenta un componente cíclico estacional.

El modelo XGBoost supera al SARIMA en términos de errores y en rapidez y peso computacional. A medida que se fueron incorporando las variables de retardo o lags, el modelo mejoraba considerablemente en resultados, muy por encima de las variables exógenas como temperatura y festividad.

El análisis de explicabilidad con SHAP y LIME evidencian que, efectivamente, el factor más influyente son las variables de retardo, concretamente el valor de la demanda eléctrica inmediatamente anterior a la predicción.

	datetime2_local	value	fest	temp_media
0	2019-01-01 01:00:00+01:00	23459.000000	1	5.007692
1	2019-01-01 02:00:00+01:00	22781.000000	1	4.480769
2	2019-01-01 03:00:00+01:00	21448.500000	1	4.094231
3	2019-01-01 04:00:00+01:00	20262.166667	1	3.751923
4	2019-01-01 05:00:00+01:00	19463.500000	1	3.440385



#### 5. Conclusiones

Se confirma que emplear un modelo complejo como XGBoost respecto de modelos tradicionales como ARIMA o SARIMA no es solo más eficiente en términos de rapidez computacional, sino que también presenta mejores resultados y mayor precisión.

La variable festividad no influyen, en comparación con las variables de retardo, en gran medida en las predicciones de demanda.

La variable temperatura influye más que la festividad y lo hace en forma de U. Temperaturas muy altas o muy bajas presentan un impacto mucho mayor en la demanda, mientras que temperaturas neutras no modifican en gran medida la predicción de la demanda.

El proyecto se alinea con los Objetivos de desarrollo sostenible, destacando su impacto energético y medioambiental.

#### 6. Referencias

[1] J. Masís, Interpretable Machine Learning with Python, 2022.

[2] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, Time Series Analysis: Forecasting and Control. Wiley, 2015.

[3] R. Harode, “XGBoost: A Deep Dive into Boosting,” Medium, 2020. [Online]. <https://medium.com/sfu-cspmp/xgboost-a-deep-dive-into-boosting-f06c9c41349>

# STATISTICAL MODELS FOR ELECTRIC DEMAND FORECASTING

**Author: Garate Aspichueta, Isabel Katalin.**

Director: del Saz-Orozco Huang, Pablo Carlos.

Collaborating Entity: ICAI – Universidad Pontificia Comillas

## SUMMARY

### 1. Introduction

This project focuses on the development of advanced statistical models (ARIMA, SARIMA, and XGBoost) to predict electricity demand in Spain. It integrates explainability techniques (SHAP and LIME) into these models to understand the factors influencing the predictions.

### 2. Definición del proyecto

The main objective of the project is to develop advanced predictive statistical models and apply explainability techniques to build a forecasting system capable of anticipating electricity consumption in the Iberian Peninsula.

To this end, an in-depth analysis of demand trends in recent years and the factors influencing consumption was first carried out. Advanced predictive models and explainability techniques were also researched.

Next, data on electricity demand and influencing variables—such as temperature, day type, and public holidays—were collected and processed for the period 2019–2022.

Three models (ARIMA, SARIMA, and XGBoost) were then trained and analyzed using comparative metrics such as RMSE and MAE.

Finally, these models were examined through the application of SHAP and LIME explainability techniques, alongside qualitative analysis.

Conclusions were drawn regarding the most influential variables in demand and how to improve model performance.

### 3. Descripción del modelo/sistema/herramienta

The project was developed in Python, using Anaconda within a Jupyter Notebook environment. Libraries such as pandas, pmdarima, XGBoost, and SHAP were employed. The pipeline includes data extraction from Red Eléctrica de España, preprocessing of the hourly time series, and extraction and transformation of features (including temperature and public holidays), followed by comparative evaluation of the models.

XGBoost proved to be the most accurate, also standing out for its explainability. SHAP enabled a global understanding of how variables influence predictions, while LIME provided more local, instance-specific explanations. Both techniques are complementary, and their results were aligned

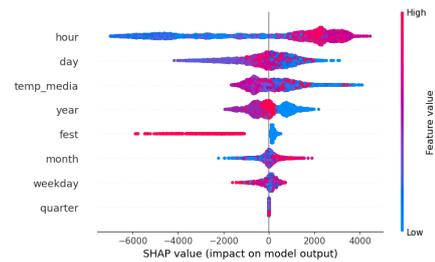
### 4. Resultados

The SARIMA model outperforms ARIMA by incorporating the seasonal component (s), which is essential in a time series with a clear cyclic seasonal pattern.

XGBoost surpasses SARIMA in terms of prediction error, speed, and computational efficiency. As lag variables were added to the model, performance improved significantly—far more than when including exogenous variables such as temperature and holidays.

Explainability analysis using SHAP and LIME confirms that lag variables, especially the immediate previous electricity demand value, are the most influential in the predictions.

	datetime2_local	value	fest	temp_media
0	2019-01-01 01:00:00+01:00	23459.000000	1	5.007692
1	2019-01-01 02:00:00+01:00	22781.000000	1	4.480769
2	2019-01-01 03:00:00+01:00	21448.500000	1	4.094231
3	2019-01-01 04:00:00+01:00	20262.166667	1	3.751923
4	2019-01-01 05:00:00+01:00	19463.500000	1	3.440385



## 5. Conclusions

Using a complex model like XGBoost is not only more efficient in terms of computational speed compared to traditional models like ARIMA or SARIMA, but it also yields better accuracy and overall results.

Public holidays do not significantly impact electricity demand predictions compared to lag variables.

Temperature has a greater influence than holidays, showing a U-shaped effect: very high or very low temperatures have a much greater impact on demand, whereas moderate temperatures have little effect.

The project aligns with the Sustainable Development Goals (SDGs), emphasizing its energy and environmental impact.

## 6. References

- [1] J. Masís, *Interpretable Machine Learning with Python*, 2022.
- [2] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*. Wiley, 2015.
- [3] R. Harode, “XGBoost: A Deep Dive into Boosting,” *Medium*, 2020. [Online]. Available: <https://medium.com/sfu-cspmp/xgboost-a-deep-dive-into-boosting-f06c9c41349>

# Índice

<b>1. Capítulo I: Introducción .....</b>	<b>13</b>
1.1 Problema de investigación .....	13
1.2 Motivación y objetivos del proyecto .....	14
1.3 Estructura del proyecto.....	15
<b>2. Capítulo II: Demanda eléctrica en España .....</b>	<b>17</b>
2.1 Conceptos fundamentales.....	17
2.2 Factores que afectan a la demanda .....	22
2.3 Patrones temporales .....	25
<b>3. Capítulo III: Estado del arte.....</b>	<b>32</b>
3.1 Fuentes de datos y su evolución .....	32
3.2 Modelos predictivos .....	34
3.2.1 Modelos ARMA, ARIMA y SARIMA.....	34
3.2.2 Métricas de diagnóstico del modelo.....	37
3.2.3 Modelo XGBoost.....	38
3.3 Métodos de explicabilidad .....	40
3.3.1 SHAP .....	42
3.3.2 LIME .....	47
3.3.3 Comparación SHAP y LIME.....	49
<b>4. Capítulo IV: Preparación de herramientas.....</b>	<b>51</b>
4.1 Preparación del entorno de desarrollo .....	51
4.2 Preparación librerías.....	51
<b>5. Capítulo V: Preparación de datos .....</b>	<b>53</b>
5.1 Datos demanda energética .....	53
5.2 Datos factores climatológicos .....	54
5.3 Datos festividades.....	58
<b>6. Capítulo VI: Modelado .....</b>	<b>62</b>
6.1 Modelos .....	63
6.1.1 ARIMA.....	63
6.1.2 SARIMA.....	65
6.1.3 XGBoost.....	70

<b>7. Capítulo VII: Resultados.....</b>	<b>77</b>
<b>8. Capítulo VIII: Explicabilidad y Análisis de resultados .....</b>	<b>81</b>
8.1 Técnicas de explicabilidad: SHAP y LIME .....	81
8.1.1 ARIMA.....	81
8.1.2 SARIMA.....	85
8.1.3 XGBoost.....	86
8.2 Resultados y conclusiones.....	89
<b>9. Capítulo IX: Estudio de viabilidad económica e impacto del proyecto</b>	<b>92</b>
9.1 Viabilidad económica .....	92
9.2 Impacto del proyecto y alineación ODS.....	94
<b>Referencias.....</b>	<b>99</b>



## 1. Capítulo I: Introducción

### 1.1 Problema de investigación

En un mundo donde la transición energética y la sostenibilidad son objetivos prioritarios y, al mismo tiempo, la demanda de energía se incrementa y las tecnologías avanzan de forma veloz, las compañías del sector energético se encuentran con retos considerables para anticipar y cubrir la demanda de energía. Este pronóstico se ha fundamentado durante años en técnicas estadísticas tradicionales. Como señalan los autores de [1], inicialmente se utilizaron métodos convencionales para pronosticar el consumo de electricidad; concretamente técnicas como el suavizado exponencial (ETS), ARIMA y regresión lineal se han aplicado ampliamente durante más de tres décadas. Sin embargo, debido al avance tecnológico y a la gran cantidad de datos, se han ido incorporando métodos más sofisticados, como el aprendizaje automático y la inteligencia artificial.

Puesto que la energía tiene restricciones considerables para su almacenamiento a gran escala, para reducir el gasto energético y asegurar el balance entre oferta y demanda del sistema, la predicción exacta de la demanda es un elemento esencial. Y, a pesar de que las tecnologías citadas posibilitan la realización de proyecciones bastante exactas, muchas de estas soluciones carecen de explicabilidad total. En otras palabras, las compañías aplican estos modelos como ‘cajas negras’, sin entender completamente cómo funcionan internamente, lo que restringe su habilidad para fundamentar decisiones y confiar plenamente en los resultados.

En cambio, contar con un modelo de explicabilidad permite no solo entender por qué se ha hecho una determinada predicción, sino también explorar cómo varía esa predicción si se modifican las variables de entrada. Por ende, si se tiene un modelo explicable, se podrán anticipar escenarios y realizar análisis de sensibilidad respondiendo a preguntas como: ¿y si mañana la temperatura baja 5°C?, ¿cuánta energía se tendrá que generar?

Esta capacidad resulta especialmente valiosa en un contexto marcado por el cambio climático, ya que permite adaptar las predicciones ante imprevistos y mejorar la toma de decisiones estratégicas.

Este proyecto se centrará, por tanto, en examinar algunas metodologías predictivas avanzadas empleadas en el sector energético y su capacidad de explicación para conseguir mejorar la precisión de las predicciones y ofrecer un entendimiento claro de los factores que afectan los resultados. Así, se logrará emplear las predicciones no solo como valores específicos, sino también como posibles datos para futuros análisis.

Finalmente, gracias a la combinación de análisis de técnicas estadísticas avanzadas y su explicabilidad, las empresas podrán optimizar sus recursos, disminuir el impacto ambiental y mantener su competitividad, a través de una planificación estratégica más rigurosa y una toma de decisiones con más información.

## 1.2 Motivación y objetivos del proyecto

Por todo ello, la motivación de este proyecto radica en explorar una clara oportunidad de mejora de los modelos estadísticos empleados en la predicción de la demanda energética, a través de un entendimiento profundo de estos y ofreciendo explicación a sus resultados.

Con ello, se busca aportar valor a las empresas del sector energético, para que puedan emplear estos métodos avanzados en la previsión para minimizar la diferencia oferta-demanda y utilizar de manera más eficiente sus recursos. Como motivación final, se considera que esto no solo tendrá un impacto positivo en las empresas del sector, sino también en la sociedad y el medio ambiente.

El objetivo principal de este trabajo es, como se ha expuesto, contribuir a un sistema energético más sostenible y eficiente, planteando interrogantes sobre modelos estadísticos que están siendo empleados hoy, y respondiendo a esos interrogantes con análisis exhaustivos. Más concretamente, los 4 objetivos a resolver son los siguientes:

1. Desarrollar modelos predictivos para la demanda energética en España: se buscará desarrollar un modelo empleando técnicas de estadística avanzada para predecir la demanda energética con precisión.

2. Analizar la explicabilidad de los modelos predictivos: valorar la explicabilidad de los modelos empleados actualmente y de otras técnicas mencionadas como SHAP y LIME (entre otras).
3. Contribuir a la transición energética: utilizar las predicciones para una mayor optimización de recursos y proporcionar herramientas para alinear las operaciones de las empresas del sector energético con los ODS.
4. Facilitar la toma de decisiones informadas a las empresas: ofrecer conclusiones claras y justificadas sobre explicabilidad de las predicciones obtenidas para que sirvan como insights a las empresas para permitirles planificar de manera más efectiva y ofrecer una mayor confianza en los modelos.

Este proyecto se alinea con tres objetivos del plan de objetivos globales de la ONU:

- Objetivo 7: Energía asequible y no contaminante. Mejorando la predicción de la demanda, se contribuye a un uso más sostenible de los recursos energéticos y a promover un uso de energía sostenible para todos.
- Objetivo 9: Industria, innovación e infraestructura. La aplicación de técnicas novedosas (o que todavía no se han empleado en este campo) fomenta la innovación y el desarrollo del sector energético.
- Objetivo 13: Acción por el clima. Buscamos alternativas más eficientes para reducir el desperdicio de energía y optimizar recursos, contribuyendo a combatir el cambio climático.

### 1.3 Estructura del proyecto

Este proyecto se ha estructurado de la manera siguiente:

Capítulo I: En el primer capítulo se introduce brevemente el tema tratado, además de dar una explicación acerca de la motivación y objetivos del proyecto. También se abordan los Objetivos de Desarrollo Sostenible (ODS) con los que se alinea el proyecto y finalmente, se presenta la estructura del mismo.

Capítulo II: Este capítulo está dedicado a un análisis de la demanda eléctrica, tanto de aquellos conceptos relacionados con la energía, como de los factores que influyen en el valor diario/horario de demanda eléctrica y patrones temporales en la curva de demanda (recogidos de la web de Red Eléctrica Española, REE).

Capítulo III: Se basa en una profundización del estado de la cuestión, por lo que primero se mencionan las fuentes de datos que se emplean para más adelante abordar los métodos actuales de predicción: ARMA, ARIMA, SARIMA, XGBoost y modelos de redes neuronales. Posteriormente se detallan, comparan y analizan en profundidad los métodos de explicabilidad SHAP y LIME.

Capítulo IV: En este capítulo se expone la preparación de herramientas para el modelado. Se divide en dos partes, en primer lugar la preparación del entorno y el lenguaje de programación. Seguidamente se detallan todas las librerías empleadas con una breve descripción de cada una.

Capítulo V: Se detallan los procesos de extracción, limpieza y pre-processamiento de datos que se han llevado a cabo para la elaboración de los modelos.

Capítulo VI: En este capítulo se aborda el desarrollo de los distintos modelos, detallando los pasos seguidos y una breve introducción a resultados.

Capítulo VII: Se analizan minuciosamente los resultados y se extraen conclusiones relevantes.

Capítulo VIII: Se aplican las técnicas de explicabilidad de SHAP y LIME y se extraen conclusiones.

Capítulo IX: Se elabora un estudio de viabilidad económica del proyecto.

Capítulo X: Últimas conclusiones y organización del proyecto.

## 2. Capítulo II: Demanda eléctrica en España

Este capítulo estará dedicado a analizar la demanda eléctrica, que es la variable dependiente de los modelos a realizar. A lo largo del capítulo, se profundizará en los conceptos técnicos básicos involucrados para entender el comportamiento de la variable dependiente, en aquellos factores que afectan a la demanda y finalmente en el análisis de estacionalidades a lo largo de los años y patrones diarios.

### 2.1 Conceptos fundamentales

Como se ha mencionado, a continuación, se analizarán aquellos conceptos fundamentales del campo de la energía para asegurar que se presenta una base sólida de conceptos técnicos necesarios para interpretar correctamente los resultados del presente trabajo. Se busca que, a través de las siguientes notas prácticas, pasando de lo general a lo concreto, no queden términos relevantes sin esclarecer para comprender el comportamiento de la variable dependiente que se busca predecir: la demanda eléctrica horaria en España.

En primer lugar, la demanda eléctrica tiene una peculiaridad respecto de la demanda de cualquier otro sector energético, como el gas o los combustibles. A diferencia de estos sectores, la electricidad se debe generar en el mismo instante en el que se consume, pues presenta limitaciones para su almacenamiento. Esto exige un equilibrio entre demanda y generación muy preciso y constante, que solo puede lograrse con predicciones de alta calidad.

Dentro del sector, se aprecia una interesante evolución descendente en los últimos años y una pequeña subida en la demanda en el último año 2024, con un pico en 2021 como se puede observar en la Figura 1. Ello es resultado de múltiples factores, como una mayor eficiencia energética en los sistemas de iluminación, aislamiento térmico y sistemas industriales. Otra causa podría ser un cambio estructural en los sectores que tradicionalmente han sido más consumidores de energía, ya que avanzamos hacia un mundo más digital y menos dependiente de la industria pesada (que en líneas generales se caracteriza por procesos más robustos y menos eficientes energéticamente). Una

causa de menor impacto pero también relacionada es el autoconsumo fotovoltaico, es decir la energía que se genera de placas solares en hogares o edificios para el autoconsumo no se registra en los datos de demanda de REE, y representa un fenómeno relativamente reciente, pero en crecimiento. Temperaturas más moderadas en años como 2023 también son una causa del descenso en la demanda eléctrica, como se explicará en el siguiente apartado. Y finalmente, un cambio en el patrón de consumo con el teletrabajo post-pandemia a partir del año 2020.

Por otro lado, resulta interesante observar la tendencia creciente hacia el uso de fuentes de energía renovables, cuya generación presenta un crecimiento del 10,3% en el año 2024 respecto de 2023, como se muestra en la Tabla 1. Se puede observar cómo, en dicho año, la generación renovable alcanza máximos históricos llegando a suponer 148.999 GWh de la demanda total. Ello implica mayor disponibilidad de energía renovable, acercándonos a los Objetivos de Desarrollo Sostenible mencionados en la introducción, constituyendo un sistema más sostenible y responsable con el medio ambiente.

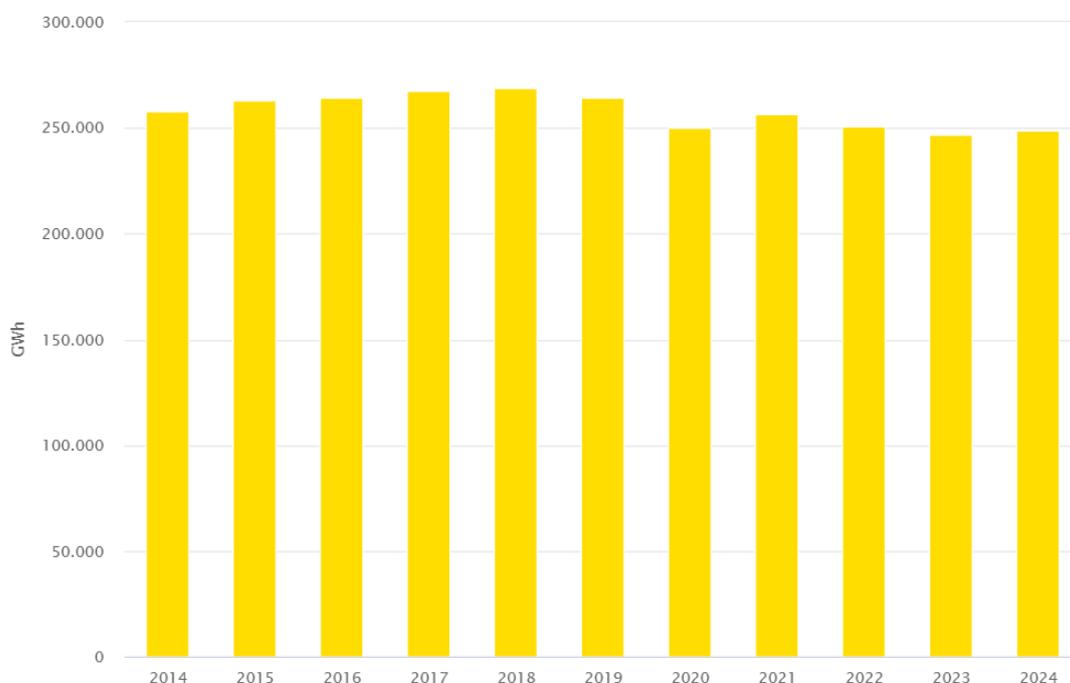


Figura 1: Evolución de la demanda eléctrica en España. (Fuente: ree.es)

	2020	2021	2022	2023	2024	%24/23
<b>Hidráulica</b>	30.632	29.626	17.911	25.765	34.912	35,5
<b>Eólica</b>	54.906	60.526	61.195	62.672	60.921	-2,8
<b>Solar fotovoltaica</b>	15.302	20.981	27.906	37.454	44.520	18,9
<b>Solar térmica</b>	4.538	4.706	4.123	4.696	4.127	-12,1
<b>Hidroeléctrica</b>	20	23	23	18	23	30,7
<b>Otras renovables</b>	4.482	4.720	4.660	3.595	3.690	2,7
<b>Residuos renovables</b>	726	878	878	846	805	-4,9
<b>Generación renovable</b>	<b>110.605</b>	<b>121.459</b>	<b>116.695</b>	<b>135.046</b>	<b>148.999</b>	<b>10,3</b>
<b>Nuclear</b>	55.758	54.041	55.934	54.647	52.391	-4,1
<b>Ciclo combinado</b>	44.023	44.500	68.137	46.682	35.755	-23,4
<b>Carbón</b>	5.021	4.983	7.762	4.006	3.030	-24,4
<b>Motores diésel</b>	2.399	2.517	2.548	2.561	2.517	-1,7
<b>Turbina de gas</b>	407	424	657	761	669	-12,2
<b>Turbina de vapor</b>	1.388	1.108	1.207	1.255	1.179	-6,0
<b>Fuel + Gas</b>	0	-0	0	0	0	300,0

Tabla 1: Evolución del consumo eléctrico en España por dividido por fuentes. (Fuente: ree.es)

Durante décadas en España, el consumo eléctrico ha estado estrechamente relacionado con la actividad económica (medida a través del PIB), especialmente en el periodo en el que España se acercaba a los estándares europeos (hasta 2008 aproximadamente). [3]

Sin embargo, desde hace unos años esta relación se ha ido desvaneciendo notablemente. [4]

Por un lado, el *Informe del Sistema Eléctrico Español de 2019* muestra que la demanda eléctrica se encuentra por debajo de la de 2008 a pesar de una recuperación del PIB, lo que evidencia esta desvinculación (Red Eléctrica de España, 2020).

Por otro lado, desde los años 80 en España el sector servicios ha ido ganando relevancia frente al sector industrial, que históricamente demandaba más energía. Según el IDAE (Instituto para la Diversificación y Ahorro de la Energía) en su informe de *Tendencias y políticas de eficiencia energética en España*, entre 2000 y 2022 la intensidad de consumo energética disminuyó un 24,7 %, en gran medida como consecuencia de la evolución del

modelo productivo del país, en el que el peso relativo de la industria ha ido descendiendo frente al crecimiento del sector servicios. Este cambio estructural implica que una parte creciente de la economía se basa en actividades menos intensivas en consumo energético, lo que contribuye a esta desvinculación entre la evolución del PIB y la del consumo eléctrico. Dicho informe también destaca la adopción de medidas de ahorro y eficiencia energética como factor influyente en este fenómeno. (IDAE, 2024)

Esta transformación implica que el crecimiento del PIB, que antes estaba estrechamente relacionado con la demanda eléctrica, ahora pierde relevancia como predictor directo. Este cambio destaca la necesidad de adoptar enfoques más flexibles en los modelos predictivos que tengan en cuenta estos cambios estructurales.

Como se menciona, es interesante analizar cómo los patrones de consumo han ido evolucionando ya que esto señala la importancia del uso de modelos más sofisticados que sean autorregresivos (es decir, que incorporan valores pasados de la variable dependiente como input). Además, también resulta fundamental aplicar técnicas de explicabilidad que permitan tanto interpretar los resultados y conocer en qué medida influye cada variable en el resultado final, como elaborar análisis de sensibilidad para anticipar resultados. Y precisamente en ahí donde se justifica el uso tanto de modelos avanzados de predicción como de herramientas como SHAP o LIME para su explicabilidad.

En este proyecto, se utilizarán los datos de demanda eléctrica real horaria de Red Eléctrica de España (REE), cuya extracción se detallará más adelante en el Capítulo V: preparación de datos. REE es el operador del sistema eléctrico peninsular, por lo que la demanda que se tendrá en cuenta representa la energía solicitada por horas de los usuarios de la península.

Los datos recogidos estarán expresados en MW (megavatios), medida de potencia instantánea: la cantidad de energía que se produce/genera en un instante, equivale a  $10^6$  vatios.

Por otro lado, la energía también se puede medir en MWh (megavatio-hora): la cantidad total de energía consumida (o generada) en un periodo de tiempo determinado. Conviene conocer la diferencia, pero en este proyecto se trabajará con los MW medidos cada hora.

Finalmente, es también interesante definir la estructura del sistema eléctrico español, para entender cuáles son los actores que intervienen y cómo funciona el proceso desde la generación al transporte, distribución y consumo. [5]

El sistema peninsular está organizado de la siguiente manera:

1. Los primeros del proceso son los productores de energía: es aquí donde empieza el proceso ya que son las empresas que se encargan de producir la electricidad. Lo hacen mediante fuentes de energía no renovables como el carbón o gas y con renovables (solar, eólica, hidroeléctrica...).

Antes de transportar esta energía, hay que determinar cuánto se va a generar, por quién y a qué precio, y esto se hace en el mercado eléctrico diario e intradiario. Este proceso lo lleva a cabo OMIE (Operador del Mercado Ibérico de Energía) en la península. En el mercado diario se hace para cada hora del día siguiente y en el intradiario sin embargo se pueden ir haciendo ajustes con menos antelación y más cerca del momento de consumo (para adaptarse a la meteorología, por ejemplo).

2. Esta energía es transportada desde los puntos de generación hasta las redes de distribución (que llevan a los consumidores finales). En esta fase de transporte hasta redes de distribución es donde entra REE, siendo un agente clave ya que actúa como transportista (único) y como gestor del sistema eléctrico. En primer lugar, el transporte debe ser lo más eficiente posible y REE lo hace mediante líneas de alta tensión, garantizando equilibrio instantáneo entre generación y consumo, lo cual es muy importante para asegurar un flujo continuo y seguro de electricidad por toda la red nacional.

3. A continuación aparecen las distribuidoras, que operan en redes de media y baja tensión para llevar la electricidad desde los centros de transformación hasta los consumidores finales.
4. Después son las empresas comercializadoras las que venden la electricidad a los consumidores, ya sea en el mercado regulado o mercado libre.
5. Al final, la electricidad llega a los hogares, industrias y servicios para su consumo.

Cuando se analiza el proceso desde la generación hasta el consumo, es fácil ver cómo la predicción es vital para que todo el ciclo funcione. Desde este punto de vista, REE necesita predicciones para planificar con antelación qué centrales tendrán que estar activas, en qué momentos, etc. Y del mismo modo, a nivel de mercado, se necesitan predicciones de calidad para poder establecer precios y ofertas en el mercado diario e intradiario.

## 2.2 Factores que afectan a la demanda

Como se ha ido anticipando, la demanda eléctrica depende de gran cantidad de variables, algunas más fáciles de predecir y otras más complejas que se tratarán como 'factores no predecibles'.

En primer lugar, los factores climatológicos. Estos tienen un impacto directo y significativo en la demanda ya que gran parte del consumo viene de los sistemas de climatización, iluminación, etc. Los principales son los siguientes.

- Temperatura: es el factor que más influye en el uso de sistemas de calefacción o refrigeración, por lo que tiene mucho impacto en la demanda. Esto se recoge en el artículo de Filonchik y Dolzhenko (2022), donde se afirma que la temperatura exterior es el principal impulsor del consumo residencial de electricidad, ya que de ella depende el uso de sistemas de calefacción o refrigeración, tanto en condiciones de frío extremo como de mucho calor. De hecho, las temperaturas extremas suelen generar picos de demanda, por lo que es el factor principal que se tendrá en cuenta en los modelos. Esta relación entre temperaturas extremas

y picos de demanda ocurre también de manera opuesta, es decir, en meses de temperaturas moderadas lo normal es encontrar una demanda menor que en meses de verano o invierno más extremos. Y esta misma relación se puede observar de manera anual, como se ha visto en el ejemplo de la Figura 1 del año 2023. [6]

De manera ilustrativa, se presenta como Figura 2 un gráfico de elaboración propia que muestra la temperatura media horaria y la demanda eléctrica correspondiente al 1 de septiembre de 2021. Este ejemplo permite observar la relación que puede existir entre estos dos factores. Evidentemente la temperatura es solo uno de los muchos factores que influyen, pero resulta interesante observar la relación en sus tendencias: picos y valles en mismas franjas horarias.

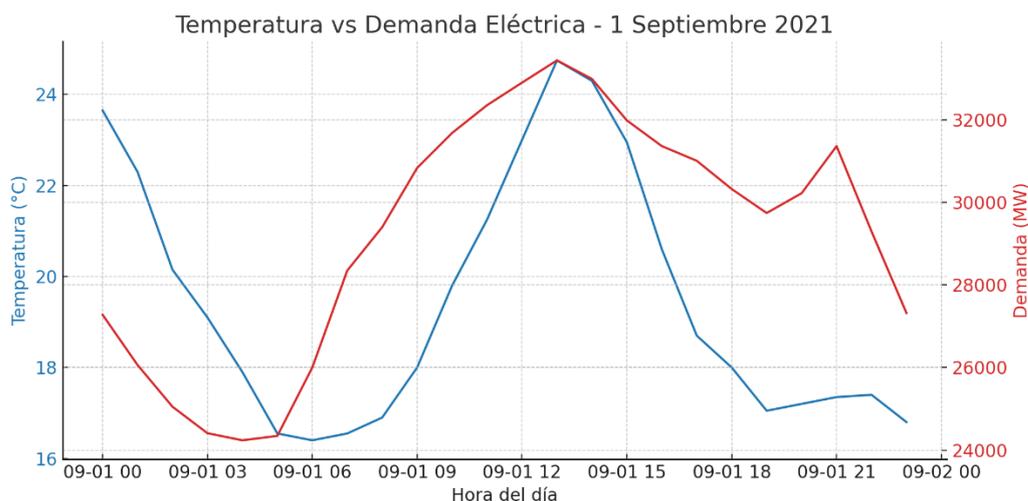


Figura 2: Demanda y temperatura diarias 1 septiembre 2021. (Fuente de elaboración propia)

Sin embargo, cabe señalar que el 1 de septiembre de 2021 fue miércoles. Por lo tanto, la correlación entre temperatura y demanda observada podría ser un patrón típico de un día laborable, en el que las temperaturas suben al mediodía y son más bajas por la noche, y la demanda crece en paralelo con el inicio de la actividad económica. Para valorar si la temperatura es un factor determinante o si los picos de consumo responden más bien a horas pico en días laborables (por ejemplo, a las 9 de la mañana), se presenta en la Figura 3 el mismo análisis para el 5 de septiembre, domingo.

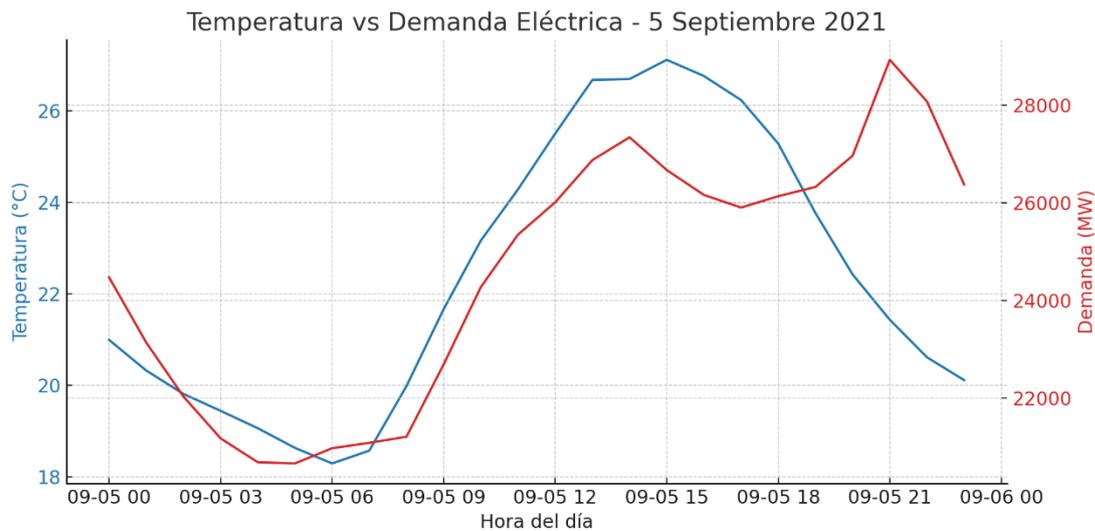


Figura 3: Demanda y temperatura diarias 5 septiembre 2021. (Fuente de elaboración propia)

Al comparar ambos días se puede observar que, modificando el tipo de día (laborable o no laborable), el comportamiento de la demanda eléctrica cambia, teniendo picos menos extremos y desplazados hacia mediodía-tarde. Esto no quiere decir que la temperatura no influya, pero evidencia la relevancia que tiene el día de la semana (si es laborable o no) en la curva de la demanda. Por ello, se expone más adelante en este mismo apartado un análisis en mayor profundidad acerca del factor ‘festividad’.

- Precipitación: aunque tiene un impacto menos directo puede modificar los hábitos de movilidad/quedarse en casa, lo que a su vez altera el consumo eléctrico.
- Viento: puede influir en la sensación térmica y, por ende, en el uso de la climatización de una forma similar a la temperatura, y además afecta directamente a la generación eólica (esto no modifica la demanda, pero sí el equilibrio oferta-demanda).
- Radiación: afecta tanto al uso de iluminación como al uso de la climatización. Es decir, días soleados en invierno podrían hacer que se reduzca la demanda en sistemas de calefacción.

- Nubosidad: puede influir en la demanda de manera inversa a la radiación. Días nublados suelen estar asociados a mayor demanda de iluminación artificial y, en invierno, mayor uso de calefacción.

Festividades: Esta variable tiene gran impacto en la demanda, como se ha visto en el análisis de temperaturas, ya que los hábitos de las personas cambian radicalmente en días festivos.

Por un lado, la actividad industrial y comercial baja, hay mayor actividad residencial y por ende también cambian los horarios, por lo que los picos de consumo pueden desplazarse a diferentes horas. Además, también puede ocurrir que su influencia se llegue a extender a días anteriores o posteriores a festivos (puentes, vísperas, etc.). Los días festivos suelen presentar hábitos similares a los fines de semana, por lo que se tratarán del mismo modo a partir de ahora (más detallado en el Capítulo V: preparación de datos).

Otros factores no predecibles: todos aquellos factores que pueden afectar a la demanda y que son difíciles de anticipar, por ejemplo, eventos excepcionales como pandemias, huelgas generales, apagones, cambios regulatorios, etc. y también cambios en el comportamiento de los ciudadanos, como el teletrabajo.

### 2.3 Patrones temporales

En este apartado se explorarán los patrones de comportamiento de la curva de la demanda eléctrica diaria, mensual y anual, muy vinculados con los factores tratados en el apartado anterior: condiciones climatológicas, horarios laborales, fines de semana y días festivos.

Para profundizar en estas relaciones, se incorpora un análisis comparativo entre años, diferenciando entre días laborables y festivos en distintos meses representativos (enero, marzo, julio y octubre). Con ello se podrán identificar patrones de estacionalidad y posibles efectos asociados al consumo energético.

A continuación, se presentan una serie de gráficos que muestran la curva de demanda promedio por hora para cada tipo de día y mes seleccionado, teniendo en cuenta los años desde 2019 a 2024. En primer lugar, la Figura 4 muestra el comportamiento típico de un día laborable de marzo (en promedio, teniendo en cuenta todos los días laborables de marzo).

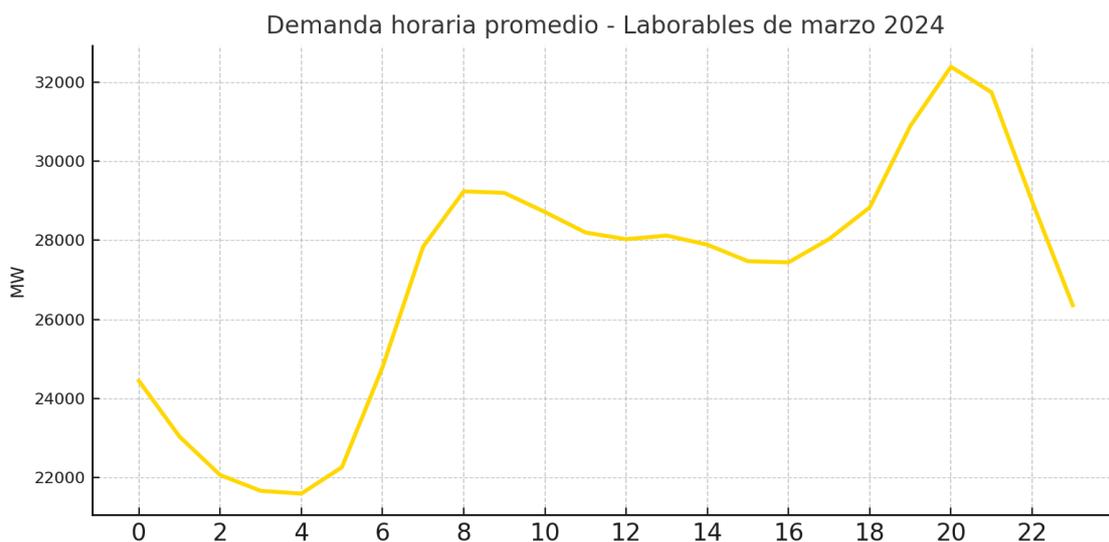


Figura 4: Curva de demanda energética diaria, laborables de marzo de 2024. (Elaboración propia)

Se observan dos picos principales:

- Pico matinal alrededor de las 8:00, coincidiendo con la primera franja horaria de jornada laboral y escolar. La demanda en este punto es de 29.000 MW aproximadamente.
- Pico alrededor de las 20:00 y 21:00, correspondiente al consumo residencial tras la jornada laboral, bastante más alto que en matinal, llegando a los 32.000 MW.

Durante la noche la demanda es baja, con un mínimo entorno a las 4:00. En las horas centrales (13:00–17:00), la demanda se estabiliza, con un ligero valle después de la hora de comer. Este tipo de curvas, como se ha ido mencionando es característica de días laborables con actividad económica regular, con picos a primera hora de la mañana y por la tarde-noche y una zona valle a la hora de comer.

Resulta muy interesante comparar este gráfico con la demanda horaria promedio, también de un día laborable y en el mismo mes, pero esta vez de 2017. Como se observa en la figura 5, el planteamiento inicial de la evolución de la demanda se confirma: en promedio, la demanda es bastante más alta en 2017 que en 2024. A pesar de que sigue la misma estructura de un pico matinal, otro vespertino y un valle entre las 14:00 y las 17:00, en el caso de la demanda de 2017, los puntos máximos llegan a alcanzar los 34.000 MW (aprox.) y el valle es mucho menos pronunciado, presentando una demanda de 30.000 MW.

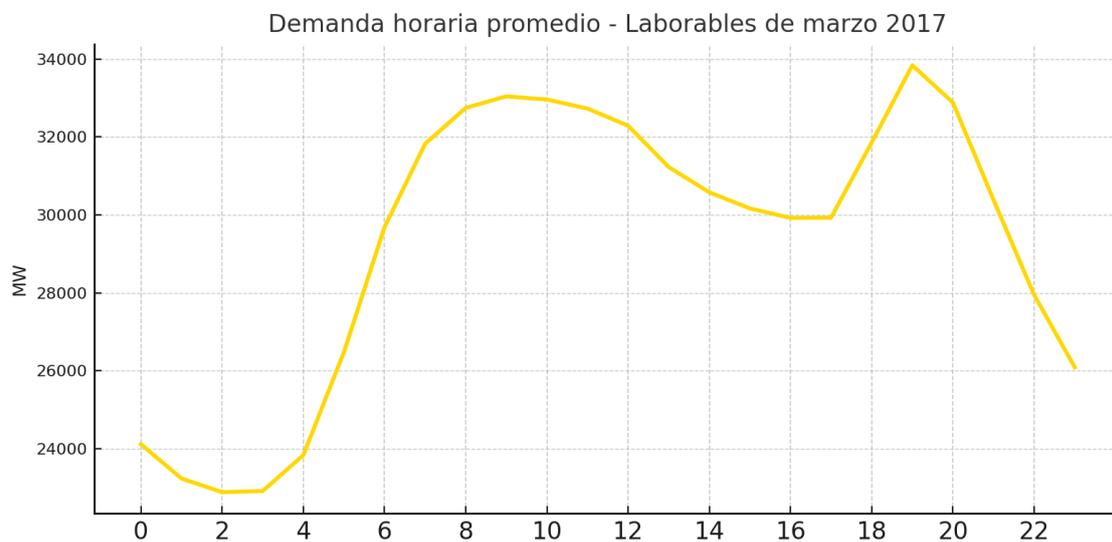


Figura 5: Curva de demanda energética diaria, laborables de marzo de 2017. (Elaboración propia)

Entrando en análisis, salta la vista en la figura 5 la amplitud del pico matinal respecto del pico observado en 2024. En este segundo gráfico, el pico matinal alcanza prácticamente la misma demanda que el nocturno. Sin embargo, en el caso de 2024, existe una diferencia considerable: el pico de las 21:00 es aproximadamente un 20% mayor que el de la mañana.

Esta notoria diferencia sugiere un cambio en los patrones de consumo, lo que podría estar vinculado al creciente uso del autoconsumo fotovoltaico mencionado en la introducción a la demanda eléctrica. Gran parte del autoconsumo fotovoltaico en el sector residencial y las empresas no se contabiliza en los datos de REE.

En 2024, la demanda durante las horas centrales del día (especialmente entre las 11:00 y las 16:00) se reduce notablemente, probablemente porque parte del consumo está siendo cubierto por la energía generada por instalaciones solares. Además, el valle es bastante más amplio y pronunciado que en 2017, lo que continúa reforzando la idea del impacto del autoconsumo fotovoltaico en la curva de la demanda eléctrica.

A continuación, se explora si existe la misma diferencia entre días festivos de marzo de 2024 y de 2017 que la que había en días laborables.

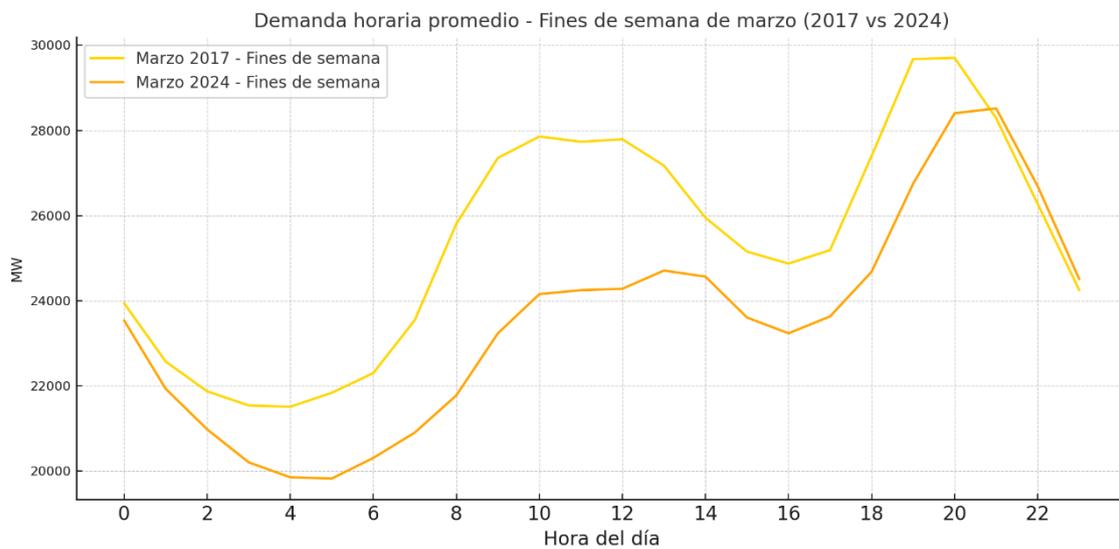


Figura 6: Comparativa de demanda promedio días no laborables marzo 2027 vs. marzo 2024

En este caso, de nuevo se observa la diferencia radical del nivel de consumo, apuntando a un uso más eficiente de la energía en 2024 que en 2017. Sin embargo, se presentan diferencias con el análisis anterior de días laborables, que pueden asociarse al autoconsumo predominante.

Mientras que en los días laborables de 2024 ya se observaba una reducción significativa del consumo eléctrico durante las horas centrales del día (11:00 a 16:00), el efecto es aún más marcado durante los fines de semana.

Esto sugiere que el autoconsumo residencial, es decir, el generado en los hogares mediante instalaciones fotovoltaicas, está desempeñando un papel predominante en la curva de demanda que el autoconsumo en empresas. En otras palabras, durante los

fin de semana, cuando la población permanece más tiempo en sus viviendas, hay una mayor diferencia entre 2017 y 2024.

Por otro lado, también se puede atribuir el cambio climático y aumento de horas de sol a esta evolución en la demanda. En relación con lo mencionado en el apartado de temperatura, si la temperatura exterior media es superior en 2024 que en 2017, no es una sorpresa que la demanda eléctrica se vea reducida. Por otro lado, y siguiendo el informe de IDEA, el hecho de que los fines de semana haya una mayor diferencia entre 2024 y 2017 podría atribuirse también a una adopción de hábitos más responsables por parte de los ciudadanos.

Este último punto también podría estar relacionado con una subida de precios que lleve a los ciudadanos a ser más conservadores con el uso de la electricidad en sus hogares y a ser menos conscientes de ello cuando se encuentran en sus puestos de trabajo. Sin embargo, los datos muestran que en marzo de 2024 el precio medio de la electricidad fue significativamente inferior al de marzo de 2017, situándose en torno a 20,3 €/MWh frente a los 52,2 €/MWh registrados en 2017 (OMIE, 2024; REE, 2017).

Para continuar con el análisis de patrones temporales, se analiza a continuación la curva de la demanda mensual, correspondiente al mes de marzo de 2024. Se ha seleccionado este mes ya que no hay (en España) extremos estacionales de invierno o verano, las temperaturas suelen ser moderadas, evitando así el uso extremo de calefacción/refrigeración, permitiéndonos observar una curva más representativa de ciclos semanales y efecto de días festivos.

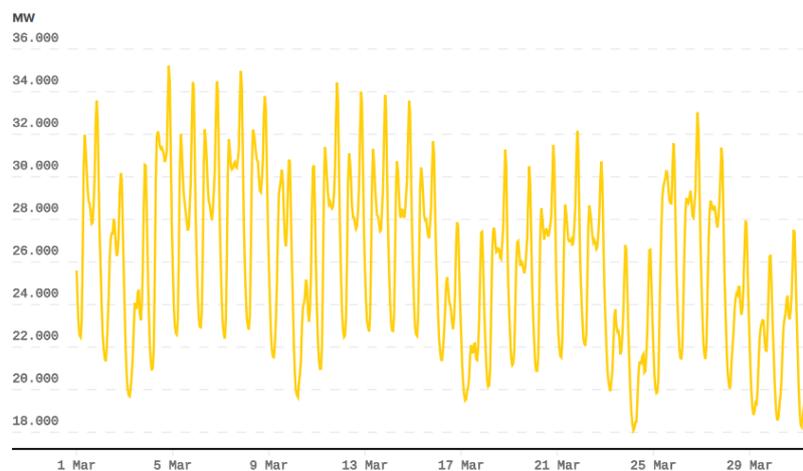


Figura 7: Curva de demanda energética mensual, marzo de 2024. Fuente: ree.es

Como se observa en la Figura 7, lo largo del mes se aprecian varios patrones destacados:

- Ciclos semanales: Se puede observar cómo, por ejemplo, del 4-10 de marzo la demanda eléctrica cae en fin de semana y tiene picos muy elevados en días laborables, patrón que se repite tantas veces como semanas en el mes.
- Además, se observan cómo los domingos (por ejemplo, 10 y 17 de marzo de 2024) la demanda presenta mínimos.
- La tendencia semanal se para el 28 de marzo, justo coincidiendo con el inicio de la Semana Santa, y a partir de ese momento su comportamiento es similar al de los domingos. Esto es un apunte que nos permitirá simplificar nuestro modelo ya que adoptaremos los días festivos como domingos.
- Existe una tendencia descendente a lo largo del mes, posiblemente por un aumento de temperaturas con la llegada de la primavera y por ende una menor necesidad de sistemas de calefacción. Esto sirve como ilustración a la hipótesis expuesta en el apartado anterior.

Finalmente, en la Figura 8 se observa la curva de la demanda eléctrica anual, correspondiente al año 2024.

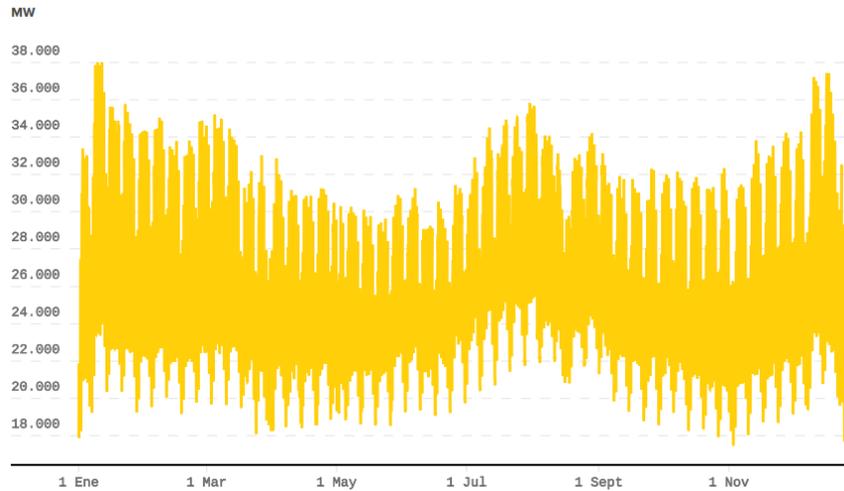


Figura 8: Curva de demanda energética anual, año 2024. Fuente: ree.es

Desde esta perspectiva, se observa de manera clara aquellos valles en los meses de temperaturas moderadas de los que se venía hablando. De manera genérica y en un vistazo rápido se pueden identificar tres estacionalidades:

- Invierno (enero, febrero y diciembre), con máximos niveles de demanda y picos diarios, asociados al uso de calefacción.
- Verano (julio y agosto), con picos por el uso de aire acondicionado.
- Primavera y otoño, meses de temperaturas moderadas que se traducen en valles en la curva de demanda anual, ya que hay menor demanda media y menor variabilidad diaria.

Este patrón anual también evidencia que el sistema energético presenta un carácter cíclico, es decir, la demanda se ajusta a las condiciones climáticas y a los ritmos socioeconómicos del país con cierta periodicidad

### 3. Capítulo III: Estado del arte

En este capítulo, se detallará la investigación realizada acerca del estado de la cuestión antes de desarrollar e implementar los modelos. Es decir, se analizará en profundidad los métodos que se emplean hoy para la predicción de demanda energética, así como los métodos actuales de explicabilidad.

Esta parte es vital para el proyecto pues servirá de punto de partida para el desarrollo de los modelos propios y de los métodos de explicabilidad empleados, además de servir como justificación de la elección de la metodología empleada.

#### 3.1 Fuentes de datos y su evolución

Por la naturaleza predictiva y estadística de este proyecto, la calidad de los datos es imprescindible para su desarrollo. La precisión de los modelos desarrollados vendrá muy determinada por la calidad de los datos, por la amplitud del horizonte temporal que cubran y por la riqueza de las variables seleccionadas.

Desde hace unos años, además, los datos de organismos nacionales e internacionales del sector son más accesibles, lo cual hace que la primera tarea de extracción de datos sea más ágil y también fomenta el desarrollo de investigaciones orientadas a la predicción de la demanda eléctrica. A lo largo del apartado, se irán analizando estas fuentes de datos y más adelante en el Capítulo IV se detallará el proceso de extracción de datos.

Red Eléctrica de España (REE): es el operador del sistema eléctrico español, por lo que la web [ree.es](http://ree.es) es la fuente principal de los datos de demanda eléctrica empleados para el desarrollo de los modelos. Desde su portal de datos hay publicada información sobre la demanda a tiempo real y también es posible acceder a datos históricos de demanda eléctrica horaria, diaria, etc.

Además, también se publican artículos en la web con datos y conclusiones valiosas para los análisis de este proyecto, los cuales se han utilizado de manera orientativa para sacar conclusiones propias y para entender qué factores son importantes analizar alrededor

del concepto de demanda eléctrica. En este trabajo se ha utilizado la demanda horaria del sistema peninsular, por ser la más representativa a escala nacional y contar con una continuidad histórica robusta.

Por otro lado, se ha optado en este trabajo por utilizar los datos de demanda en barras de central. Esto quiere decir que los datos de demanda que se han tenido en cuenta de la web de REE son los medidos para la operación del sistema, sin incluir pérdidas ni autoconsumo. Son los valores de energía que debe generarse para abastecer el sistema eléctrico sin considerar pérdidas en la red de transporte y distribución.

La otra alternativa es utilizar la de contador cliente, seleccionando como punto de referencia el punto de consumo, lo que sería más exacto al hablar de demanda ya que representa el consumo real. Sin embargo, podría estar afectado por el autoconsumo o la eficiencia del sistema y además emplear los datos de REE facilita el acceso a históricos homogéneos, por lo que se selecciona la primera opción.

AEMET OpenData: AEMET es la agencia estatal de meteorología, que en su web OpenData ofrece una API pública para acceder a los datos meteorológicos históricos. Se deben consultar bajo autorización, por lo que se debe hacer una solicitud. En concreto, los datos diarios son públicos y se pueden descargar directamente, pero para los datos horarios se debe completar un formulario y hacer una solicitud formal.

NASA Power: Pone a disposición de cualquier usuario de forma gratuita series históricas de múltiples variables climáticas. Lo hace a través de su plataforma NASA Prediction Of WorldWide Energy Sources, donde solo hay que indicar la latitud y longitud de los puntos de interés y devuelve archivos de datos de cualquier formato a elegir.

CalendariosLaborales.com: Se emplea para tener en cuenta los festivos de cada provincia, pues antes de elaborar modelos es necesario tenerlo en cuenta. La extracción de estos datos se ha automatizado utilizando Selenium, lo que permite integrarlos fácilmente en los conjuntos de entrenamiento y prueba de los modelos.

Webs de programación de modelos: Además de las fuentes de datos, se han consultado diversas plataformas para la implementación de modelos predictivos, como XGBoost y SHAP, a través de sus respectivas documentaciones oficiales. Estas herramientas han sido fundamentales tanto para el entrenamiento de los modelos como para el posterior análisis de explicabilidad de las predicciones.

## 3.2 Modelos predictivos

Como se ha expuesto en la introducción, los avances tecnológicos en aprendizaje automático y estadística han permitido aplicar una gran variedad de herramientas para elaborar modelos predictivos en múltiples ámbitos, desde predicciones en el precio del oro, hasta el tema central de este proyecto: predicción de la demanda energética. Ello implica que se han empleado tanto métodos tradicionales de modelos temporales como otros modelos más avanzados de machine learning. A continuación, se explorarán ambos.

### 3.2.1 Modelos ARMA, ARIMA y SARIMA

Dentro de los modelos más tradicionales (y algo más sencillos) se encuentran los modelos estadísticos de series temporales. Estos son los modelos ARMA, ARIMA y SARIMA, que son explicados a continuación:

El primero es el modelo ARMA, Autoregressive Moving Average, que consta de dos partes: una parte autorregresiva (AR) y otra parte de media móvil (MA). Esto significa que para predecir los valores futuros usa tanto valores pasados (parte autorregresiva) como errores pasados de las predicciones anteriores (parte de media móvil). De esta forma el modelo se puede ir ajustando. La única limitación es que este modelo solo puede emplearse en series estacionarias (la media y varianza no cambian respecto al tiempo).

Para modelar series que no son estacionarias se utiliza el modelo ARIMA, Autoregressive Integrated Moving Average, que deriva del ARMA solventando dicha limitación. Este

modelo, según Box y Jenkins en su libro 'Time Series Analysis: Forecasting and Control' sigue una metodología sistemática por pasos:

1. El primer paso es la identificación del modelo, detectando que no es estacionario, para diferenciar la serie. La serie se va diferenciando para ir eliminando fuentes de variaciones no estacionarias. Por ejemplo, si la media va cambiando con el tiempo se aplica una o más diferencias para llegar a una serie estacionaria.
2. Después viene la estructura del modelo. En los modelos ARIMA la estructura es  $p, d, q$ . Siendo  $p$  el número de términos autorregresivos (normalmente de 0 a 3),  $d$  el número de diferenciaciones necesarias para lograr llegar a una serie estacionaria (normalmente 1 o 2 si con una diferencia no se elimina la tendencia) y  $q$  el número de términos de media móvil. En el libro citado, los autores mencionan que hay que tener cuidado de no asignar una  $d$  muy alta ya que puede generar modelos ineficientes e inestables.

La ecuación general después de aplicar  $d$  diferencias a la serie es la siguiente:

$$\Phi(B)(1 - B)^d X_t = \Theta(B)\varepsilon_t$$

*Ecuación 1: Ecuación general modelo ARIMA*

Siendo:

$X_t$  la serie temporal original que se busca modelar

$d$  el número de diferenciaciones necesarias para que la serie sea estacionaria

$B$  el operador de retardo, que hace que  $BX_t = X_{t-1}$

$\Phi(B)$  el polinomio que representa la parte autorregresiva (AR)

$\Theta(B)$  el polinomio que representa la media móvil (MA)

$\varepsilon_t$  el ruido blanco (error blanco), que se asume que tiene media cero, varianza constante.

Es decir, se asume que se puede aproximar por una normal de media cero y varianza constante.

1. El tercer paso es la estimación de parámetros ( $\Phi$  y  $\theta$ ) que se puede realizar aplicando métodos como máxima verosimilitud o el método de los mínimos cuadrados (se explican a continuación en el siguiente apartado).
2. A continuación, tocaría realizar el diagnóstico, es decir comprobar si los residuos tienen media cero, varianza constante y si son ruido blanco y no correlacionados.

Finalmente, el modelo SARIMA es el más genérico, ya que permite modelar series con patrones estacionales, como indica su nombre Seasonal ARIMA. Es decir, es capaz de abordar una serie que presente tanto tendencias como estacionalidad (ya sea día, mes, año...). Mantiene las tres componentes del ARIMA ( $p$ ,  $d$ ,  $q$ ) porque utiliza también la diferenciación y además incluye la variable  $s$ , periodicidad de la estacionalidad. Por ejemplo, en el caso analizado de la demanda si quisiéramos tener en cuenta la periodicidad diaria de la curva asumiendo que todos los días se repetiría el mismo patrón, sería 24 (porque los datos son horarios).

Se expresa como SARIMA ( $p$ ,  $d$ ,  $q$ ) ( $P$ ,  $D$ ,  $Q$ ) [ $s$ ], donde los parámetros en minúsculas representan las componentes no estacionales del modelo ARIMA y las mayúsculas las componentes estacionales. Por ejemplo,  $p$  sería el número de términos autorregresivos no estacionales y  $P$  el número de términos autorregresivos estacionales.

Sin embargo, el modelo SARIMA sí que presenta limitaciones cuando hay muchas estacionalidades en la serie, por lo que en esos casos se debería recurrir a modelos más complejos.

Por ende, se puede extraer que, dependiendo con qué tipo de serie se está trabajando, será conveniente emplear un modelo u otro. En resumen,

- para series estacionarias y simples, ARMA.

- para series que no presentan estacionalidad y no son estacionarias (presentan algún tipo de tendencia), será conveniente utilizar el modelo ARIMA.
- para series que presenten estacionalidad clara y leve tendencia, SARIMA.

### 3.2.2 Métricas de diagnóstico del modelo

Una vez el modelo está entrenado, se puede y debe comprobar su calidad. Esto se hace a través de métricas que permiten comparar resultados entre modelos diferentes. A continuación, se detallan las más empleadas y las que se utilizarán a lo largo del proyecto.

En primer lugar, AIC (Akaike Information Criterion), una medida de calidad relativa que lo que hace es penalizar la complejidad del modelo.

Se mide como  $AIC = 2k - 2\ln(L)$  siendo  $k$  el número de parámetros estimados y  $L$  el valor máximo de la función de verosimilitud del modelo. La función de verosimilitud indicará cuánto se alinean las predicciones a los datos observados: cuanto más alta, mejores serán las predicciones. Es decir, que a la hora de comparar dos modelos interesa que tenga mínimo AIC.

El MAE (mean Absolute Error, error absoluto medio) es una medida de error que calculará la diferencia entre la predicción y la realidad, midiendo el error de la predicción. Los errores se suman en valor absoluto para evitar que se anulen unos con otros. La fórmula es la siguiente:

$$MAE = \sum \frac{1}{n} | y_t - \widehat{y}_t |$$

*Ecuación 2: Ecuación para cálculo de MAE*

Siendo:

$y_t$  : valor real en el tiempo  $t$

$\widehat{y}_t$ : valor predicho en el tiempo  $t$

n: número total de observaciones

RMSE, o error cuadrático medio, es una medida similar al MAE pero que suma los errores elevándolos al cuadrado y después hace la raíz cuadrada de la suma, de manera que garantiza que los errores no se anulan unos con otros y así se obtiene una medida más precisa del error de predicción. Además, penaliza más los errores al elevarlos al cuadrado, por lo que tiene mayor sensibilidad a los outliers.

$$RMSE = \sqrt{\left(\frac{1}{n} \sum_{t=0}^N (y_t - \hat{y}_t)^2\right)}$$

*Ecuación 3: Ecuación para cálculo de RMSE*

También se realizará un análisis de residuos cualitativo para analizar cuándo y dónde hay errores en el modelo para detectar los motivos.

Para detectar sobreajustes, se evaluará la diferencia de MAE y RMSE entre los sets de entrenamiento con los de validación, observando si existen grandes diferencias ya que un modelo que funcione bien en entrenamiento, pero mal en la validación significará que no puede ser generalizable.

### 3.2.3 Modelo XGBoost

El modelo XGBoost, recibe su nombre de eXtreme Gradient Boost. El apodo hace alusión a que este modelo optimiza el proceso de Gradient Boosting. Se emplea por su velocidad, eficacia y capacidad para trabajar con grandes volúmenes de datos.

El Boosting se basa en agrupar secuencialmente árboles de decisión ‘débiles’ (poco profundos), haciendo que cada árbol corrija los errores que quedan de sumar las predicciones anteriores. Es decir, mezcla modelos poco profundos (pero mejores que el simple azar) para acabar creando uno fuerte. El objetivo es evitar el sobreajuste, típico

en árboles de decisión, a través de ir refinando y ajustando el modelo para crear modelos más robustos.

El Gradient Boosting se diferencia del Boosting por su factor de optimización basado en gradientes. En cada iteración, no solo añade nuevos modelos débiles, sino que estos se entrenan para aproximar el gradiente negativo de la función de pérdida,  $L(y, f(x))$ . Se puede expresar matemáticamente:

$$g_t(x) \approx -\left[\frac{dL(y, f_{t-1}(x))}{df_{t-1}}\right]$$

*Ecuación 4: XGBoost*

Siendo  $g_t(x)$  el árbol que se ajusta para que aproxime el gradiente negativo de la pérdida respecto al modelo previo.

XGBoost destaca dentro de los modelos de la familia de Gradient Boosting porque maximiza la precisión y la eficiencia en el entrenamiento a través de varias mejoras, que han sido extraídas de la publicación *XGBoost: A Scalable Tree Boosting System* (Chen & Guestrin, 2016):

1. XGBoost cuenta con un mecanismo de regularización que penaliza la complejidad del árbol. Lo hace controlando la cantidad de hojas y peso de sus nodos.

$$\sum L(y_i, \widehat{y}_i) + \sum_{k=1}^t \Omega(g_k), \quad \Omega(g) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2$$

*Ecuación 5: Función regularización XGBoost*

Añade el término mostrado en la ecuación 5 al objetivo, donde  $T$  es el número de hojas y  $w$  sus pesos.

2. Tiene también un mecanismo de optimización de segundo orden: utiliza tanto la primera derivada (el gradiente, ecuación 4) como la segunda, el Hessiano, lo que acelera y estabiliza la convergencia.

3. Su estructura de columnas facilita la lectura de resultados a través de histogramas y además optimiza el uso de almacenamiento en memoria columnar, por lo que se puede emplear con grandes volúmenes de datos. Esto hace también que se reparta el trabajo sin cuellos de botella, por lo que el entrenamiento del modelo también resulta más rápido.
4. Destaca también por cómo gestiona los datos faltantes. El propio algoritmo va aprendiendo si un valor nulo debe desviarse a la rama izquierda o a la derecha según la ganancia de información, por lo que el procesamiento de datos es mucho más simple y acaba mejorando también la robustez del modelo.
5. Finalmente, va ‘podando’ el árbol de manera automática cuando aprende que hay ramas que no aportan un incremento significativo, manteniendo solo las que mejoran la métrica. Esto hace que los modelos XGBoost sean compactos y se centren en los aspectos más relevantes de los datos.

### 3.3 Métodos de explicabilidad

En muchas ocasiones, los modelos más avanzados son precisamente los que mejores predicciones hacen, lo que implica que sus resultados son incluso más complejos de interpretar. Esto sucede sobretodo con los modelos de Inteligencia artificial (IA), que cada vez se utilizan más para predicciones. Como se ha mencionado en la introducción, estos modelos son empleados a menudo por grandes instituciones y profesionales como cajas negras, sin la capacidad de interpretar por qué el modelo hace una predicción concreta por su alta complejidad. El uso de estas técnicas incrementa la transparencia y por ende la confianza en la IA, facilitando su implementación en áreas como la salud y finanzas, a través de algoritmos. Con ello surge el concepto de Inteligencia Artificial eXplicable (IAX).

Antes de profundizar en los métodos, conviene realizar una aclaración acerca de los términos explicabilidad e interpretabilidad. Estos dos términos, a menudo empleados como sinónimos, realmente presentan diferencias en su significado. Para ello, se recurre

al libro *Interpretable Machine Learning with Python*, que ofrece una buena explicación acerca de esta diferencia.

Por un lado, interpretabilidad hace referencia a la capacidad de los humanos para entender, en un modelo de machine learning, la relación causa-efecto y la de entrada-salida. Según este libro, un modelo interpretable es aquel cuya relación entre entrada y salida es entendible por los humanos.

Por otro lado, la explicabilidad engloba todo lo anterior y va más allá. En este caso, un modelo explicable es entendible a nivel de funcionamiento interno y proceso de entrenamiento por un humano. Es decir, es mucho más transparente e implica proporcionar explicaciones sencillas y de lenguaje humano sobre el proceso interno.

Estos dos términos están muy relacionados, pero no son idénticos. Es relevante conocer las diferencias y observar cómo la explicabilidad es un término más amplio y que requiere de mayor transparencia. Sin embargo, a lo largo de este proyecto, se emplean ambos términos como parte de los esfuerzos por dotar de mayor transparencia a los modelos predictivos, indistintamente.

Estos métodos de explicabilidad se dividen en varios tipos, dependiendo de la clasificación que se haga.

Se pueden clasificar en técnicas agnósticas y específicas, dependiendo de si la aplicabilidad de la herramienta de explicabilidad depende o no del modelo de machine learning que se haya empleado. Las técnicas agnósticas son aquellas completamente independientes del modelo utilizado, aplicables a cualquiera. Por otro lado, las técnicas específicas, son las que solo pueden ser empleadas para un modelo concreto.

Otra clasificación útil es aquella que distingue si los métodos de explicabilidad aportan explicaciones locales (de un cierto parámetro, por ejemplo) o globales, que aportan un entendimiento más general del efecto de las entradas/salidas del modelo. En este caso se habla de técnicas locales y técnicas globales.

Para ello, en este apartado se va a profundizar en dos métodos de explicabilidad, SHAP y LIME, que son también conocidos como herramientas de interpretabilidad. El objetivo es llegar a entender por qué y cómo estos modelos complejos llegan a ciertas conclusiones con el fin de que este conocimiento pueda servir a las empresas como inputs futuros. La aplicación de estos modelos devolverá una medida de contribución con signo. Es decir, que devolverá un valor positivo si la característica hace que la predicción tienda a subir, y un valor negativo si lleva a la predicción a resultados menores.

### 3.3.1 SHAP

En primer lugar, uno de los métodos de explicabilidad más comunes es SHAP, de sus siglas Shapley Additive Explanations, una herramienta que se basa en la teoría de juegos para interpretar los resultados de los modelos de aprendizaje automático. Este método es cada vez más popular ya que es capaz de aportar explicaciones tanto locales como globales sobre las decisiones que toman los modelos. Es decir, es una técnica tanto global como local.

De hecho, es curioso observar que los artículos más antiguos en los que se profundiza en este método son de 2020, apenas de hace 5 años. Referenciándose unos a otros se puede llegar a trazar una red de artículos o papers escritos acerca del método SHAP con la ayuda de la web: Connected Papers. En la Ilustración 6 se puede observar esta red, en la que la mayor parte de los artículos han sido escritos en 2023 o 2024, evidenciando así la creciente popularidad y la novedad del concepto estudiado en este capítulo.

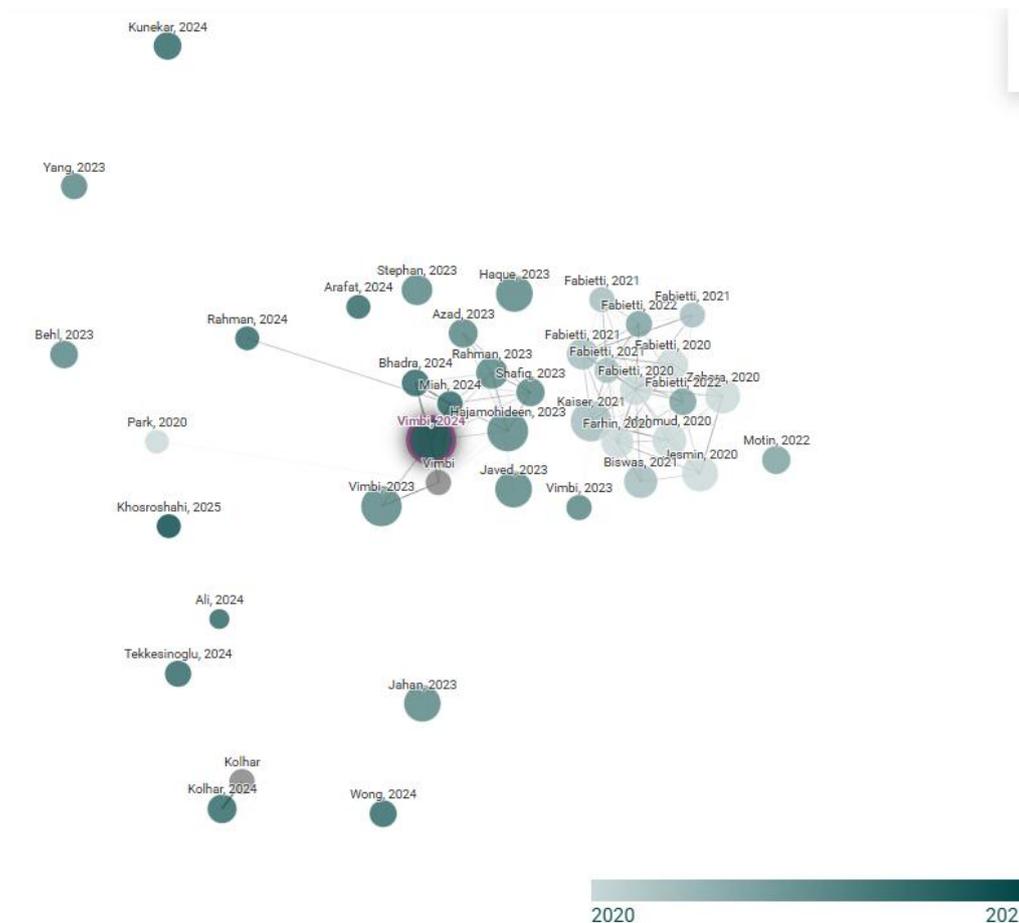


Figura 9: Red de artículos acerca de SHAP por años y referencias (Elaboración propia con la web Connected Papers)

La manera concreta en la que este método es capaz de interpretar los resultados de las predicciones es aplicando los principios de la Teoría de Juegos, un estudio que se basa en el análisis de situaciones competitivas en la que cada jugador toma una decisión que impacta al otro jugador. Es decir, la teoría de juegos analiza situaciones en las que el resultado no solo depende de la decisión de un individuo sino de la de otro u otros también.

Uno de los dilemas más famosos dentro de la teoría de juegos es el dilema del prisionero, en el que dos personas son interrogadas para ir a la cárcel. Cada uno tiene que decidir si confesar (traicionando al otro) o no confesar. Cada una de estas decisiones tiene un output diferente que viene condicionado por la respuesta del otro prisionero, haciendo difícil la decisión ya que cada uno quiere minimizar sus propias posibilidades de ir a la cárcel.

La dificultad de esta decisión radica principalmente en que los dos prisioneros no saben lo que va a hacer el otro. Desde esta perspectiva, surge una rama dentro de la teoría de juegos que es la teoría de juegos cooperativos.

En este caso, la teoría de juegos cooperativos se basa en situaciones en las que, como antes, dos (o más) individuos toman decisiones que impactan a todos los jugadores, con la diferencia de que en este caso, los jugadores pueden formar alianzas y colaborar entre sí. De esta manera, pueden formar una coalición y buscar un beneficio conjunto.

Dentro del estudio de teoría de juegos cooperativos, surge una pregunta interesante. Cuando los jugadores cooperan para alcanzar ese beneficio conjunto, y lo consiguen, pero algunos jugadores han contribuido más a esa coalición, ¿cómo se reparten los beneficios totales que se han conseguido gracias a la cooperación?

Para responder a la pregunta, Lloyd Shapley en 1953 propone el valor de Shapley, que es una manera justa de distribuir dicho beneficio entre los jugadores.

Como curiosidad, el valor Shapley viene dado por la ecuación 4 y representa la cantidad que cada jugador ( $i$ ) del grupo de  $n$  jugadores debería recibir en función de su contribución a la cooperación.

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|! (n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S))$$

*Ecuación 4: Expresión del valor Shapley. (Wikipedia)*

Donde:

$\Phi_i(v)$  es el valor de Shapley del jugador  $i$ .

$N$  es el conjunto total de jugadores (por ejemplo, todas las variables en un modelo).

$S \subseteq N \setminus \{i\}$  son todos los subconjuntos posibles de jugadores que no incluyen al jugador  $i$ .

$|S|$  es el tamaño del subconjunto  $S$ . Indica el número de jugadores.

$$\frac{|S|! (n - |S| - 1)!}{n!}$$

Es el coeficiente de ponderación.

$v(S)$  Es la ganancia de la coalición  $S$  sin el jugador  $i$

$v(S \cup \{i\})$  es el valor de la coalición al añadir el jugador  $i$  a  $S$ .

Por lo que  $v(S \cup \{i\}) - v(S)$  es la contribución marginal del jugador  $i$  a la coalición.

Para calcular el valor Shapley con la ecuación 4 del jugador  $i$  se consideran, en primer lugar, todas las posibles coaliciones que se podrían hacer con dicho jugador, se calcula su contribución marginal en cada caso y finalmente se realiza la media.

El valor Shapley, por otro lado, es perfectamente aplicable al contexto de los modelos de machine learning, donde el objetivo es calcular la aportación de cada variable a la predicción final.

Por ende, utilizando el valor Shapley como mecanismo de distribución justa, surge el método SHAP. Traduciendo al lenguaje estadístico, el juego sería la predicción del modelo y cada jugador sería una variable de entrada del modelo, que acaba influyendo en la predicción del mismo modo que la decisión de un jugador impactaba al output total. Finalmente, el valor Shapley ahora se convierte en el impacto que genera cada variable en la predicción total.

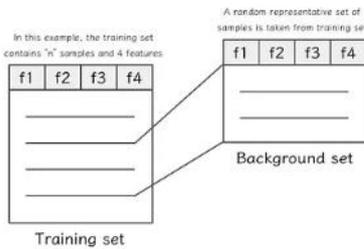
SHAP aplica este valor para medir la contribución de cada variable a una predicción determinada. Por ejemplo, si un modelo prevé que un cliente dejará un servicio, SHAP puede revelar qué factores (edad, ingresos, historial de uso, etc.) han impulsado dicha predicción hacia una probabilidad más o menos alta aplicando el valor Shapley.

A continuación, en la Ilustración 7, se expone una descripción gráfica del proceso que realiza SHAP para calcular todos los valores Shapley para una predicción  $X$  (aquello que queremos explicar).

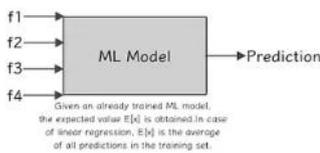
Say we want to explain feature "x":

$$X = \begin{matrix} f1 & f2 & f3 & f4 \\ \hline & & & \end{matrix}$$

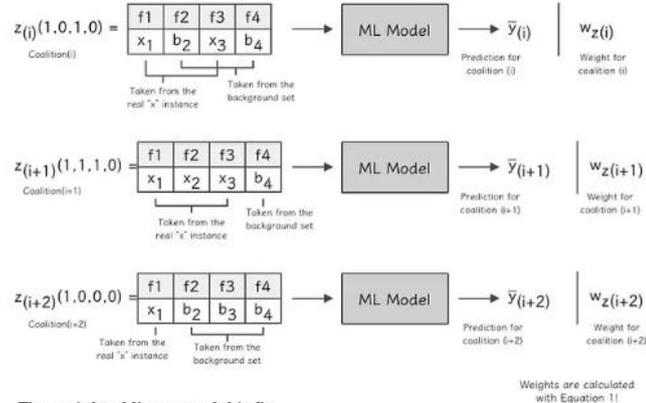
We have training set and background set:



And of course, an already trained ML model:



Coalitions, predictions and weights are calculated:



The weighted linear model is fit

With coalitions, predictions and weights, the weighted linear model is built.

Shapley values are obtained!

Once optimized the weighted linear model, the coefficients are the Shapley values!

Figura 10: proceso descriptivo de SHAP para obtener valores Shapley a partir de un modelo de Machine Learning. (Autor Tute Gomez E para Medium)

Esta aproximación que sustituye al proceso largo y tedioso de sacar los valores Shapley se denomina Kernel SHAP, uno de los métodos dentro de la técnica SHAP que además resulta ser una técnica agnóstica. Y como esta, existen muchas otras, cada una con sus características concretas.

En conclusión, SHAP ofrece explicaciones de alto nivel, consistentes y comparables. Esta consistencia y riqueza de información hace que SHAP sea un modelo muy estable para explicar predicciones. Además, en muchas ocasiones SHAP actúa como 'lenguaje común' para comparar y razonar sobre distintas técnicas de explicabilidad, unificando muchos métodos como LIME o DeepLIFT. Como posible desventaja, el cálculo de valores Shap exactos puede ser complejo y tedioso para modelos genéricos, pero las aproximaciones como KernelSHAP mencionada e ilustrada anteriormente o métodos específicos suelen mitigar esta desventaja en la mayoría de los casos y hacer posible emplear este método en la práctica.

Se expondrá más detalladamente en el Capítulo VI la aplicación de este método con Python. La implementación resulta relativamente sencilla ya que existe una librería SHAP en Python de la que se hará uso.

### 3.3.2 LIME

Otro de los métodos de explicabilidad más conocidos y extendidos en los últimos años es LIME, de sus siglas Local Interpretable Model-agnostic Explanations. Este modelo fue propuesto por primera vez en 2016 por Ribeiro, Singh y Guestrin en: *Why should I trust you?*

Su nombre resalta sus características, indicando que es una técnica local y agnóstica. Es decir, que las explicaciones que presenta no servirán como entendimiento global del modelo, sino que serán más bien explicaciones locales para predicciones individuales. Ello quiere decir que busca responder por qué un modelo predice un resultado determinado para cada caso específico, identificando así las variables que intervienen a la predicción y en qué medida lo hacen. Por otro lado, y como se ha anticipado, el término agnóstica hace referencia a la aplicabilidad del modelo, indicando que puede emplearse como técnica de explicabilidad independientemente del modelo de machine learning con el que se esté trabajando.

Al igual que SHAP, LIME se emplea con el objetivo de interpretar las predicciones de modelos complejos de machine learning, permitiendo comprender las decisiones de estos modelos y aplicarlos de manera consciente. Sin embargo, la metodología es muy diferente: mientras que SHAP usa la teoría de juegos, LIME se basa en aproximaciones locales e interpretables. Es decir, la idea es aproximar el modelo real (complejo) a través de un modelo sustituto que pueda ser interpretable, entrenándolo alrededor de la instancia (la cosa concreta) que se trata de explicar.

El proceso seguido para llegar a la interpretación de las predicciones con el método LIME se detalla a continuación.

En primer lugar, LIME genera muestras locales, creando varias versiones a modo de copias modificadas de la instancia original. Estas modificaciones se denominan versiones

perturbadas. En sets de datos similares al trabajado en este proyecto, esto se traduce en una pequeña variación numérica alrededor de un valor determinado. En textos, LIME genera variaciones cambiando ciertas palabras y en imágenes, se activan o desactivan unas regiones de la foto llamadas superpíxeles para crear diferentes copias de la imagen original.

A continuación, LIME evalúa el modelo original sobre cada una de las variaciones, obteniendo la predicción que haría el modelo para cada una de ellas.

El tercer paso es la ponderación por proximidad, es decir, se asigna un peso a cada instancia modificada que depende de cuánto se parezca a la instancia original. Es decir, que las muestras más similares influyen más en el modelo.

Finalmente, con esas muestras y sus pesos, LIME entrena un modelo interpretable. Esto quiere decir que, con todas esas versiones modificadas y los respectivos pesos, LIME entrena un modelo sencillo, como por ejemplo una regresión lineal con pocas variables, centrándose solo en el entorno cercano a la instancia que buscamos explicar.

A continuación, un ejemplo para simplificar el concepto.

Se tiene un negocio por suscripción y se busca entender por qué un cliente concreto cancela la suscripción, poniendo al cliente como una mujer de 40 años con ciertos ingresos y hábitos. LIME generaría versiones similares a la clienta presentada, por ejemplo, una con 41 años, otra con un poco más de ingresos, otra que salga a correr... Después miraría qué predice el modelo con cada uno de estos cambios y a partir de ahí puede sacar conclusiones acerca de los factores que llevan al modelo a tomar esa decisión en este caso concreto.

En conclusión, LIME resulta una técnica sencilla, que no requiere de cálculos tan complejos como SHAP, lo cual la hace también más rápida. Además, es más fácil de aplicar a modelos de diversa índole que SHAP, puesto que su metodología se basa en la simplificación del modelo, sin necesidad de acceder a los parámetros internos del

modelo. Es especialmente útil para fases exploratorias o a la hora de comunicar con usuarios no técnicos.

Sin embargo, también presenta ciertos inconvenientes en su uso. Principalmente, la inestabilidad. Al ser una aproximación local, depende del muestreo aleatorio, lo que significa que no puede ser única ni exacta y ello hace que ejecuciones diferentes puedan llevar a destacar atributos ligeramente distintos. Además, al enfocarse en las variables más influyentes, la suma de sus aportaciones no siempre coincide con la predicción inicial del modelo.

### 3.3.3 Comparación SHAP y LIME

Como último punto, resulta interesante realizar una tabla comparativa con aquellas características relevantes de cara a la aplicación de estos dos métodos de explicabilidad.

A continuación, en la Tabla 2, quedan recogidas estas características, además de una breve explicación de las particularidades de cada modelo.

Característica	SHAP	LIME
<b>Tipo de explicación</b>	Local y global	Local
<b>Metodología</b>	Teoría de juegos (sin entrenar el modelo)	Entrena el modelo interpretable
<b>Compatibilidad</b>	Agnóstico	Agnóstico
<b>Facilidad de uso</b>	Mayor coste computacional	Más simple, más rápido y sencillo de implementar
<b>Estabilidad</b>	Alta estabilidad	Media, puede ser inestable
<b>Interpretabilidad</b>	Alta, visualmente es más complejo	Muy intuitiva

<b>Aplicación práctica</b>	SHAP library	
----------------------------	--------------	--

*Tabla 2: Comparativa entre modelos SHAP y LIME. (Elaboración propia)*

Un enfoque que enriquece más aun las interpretaciones de las predicciones, según el libro *Interpretable Machine Learning with Python*, es emplear los dos modelos para que se complementen entre ellos y obtener mejores resultados. Por un lado, LIME para obtener explicaciones locales e intuitivas y predicciones individuales y por otro SHAP, para generar explicaciones tanto locales como globales con mayor estabilidad y respaldo teórico.

Aunque estas técnicas han ido ganado reconocimiento en campos como la salud y las finanzas, como se ha mencionado, su uso en el sector energético todavía es muy reciente. Esto supone una gran oportunidad para explorar el papel que podrían llegar a tener en la mejora de transparencia y fiabilidad de los modelos.

## 4. Capítulo IV: Preparación de herramientas

La preparación de herramientas y del entorno Python donde se trabajará es un paso crucial para el desarrollo de este proyecto de investigación, pues es la forma de garantizar la eficiencia y calidad del trabajo.

A continuación, se detallarán todas las aplicaciones, librerías, técnicas y recursos que han sido empleados para el desarrollo de los modelos, la implementación de técnicas de explicabilidad y el posterior análisis de resultados.

La estructura será de lo general a lo concreto, comenzando por el lenguaje de programación y el entorno de desarrollo, y continuando con librerías.

### 4.1 Preparación del entorno de desarrollo

El lenguaje de programación elegido ha sido Python a propuesta del tutor por su sencillez, capacidad para manipular grandes volúmenes de datos y la amplia variedad de librerías disponibles útiles para el propósito del proyecto.

Para el desarrollo del proyecto en lenguaje Python se ha empleado JupyterLab, una herramienta que ha permitido editar en notebooks, de manera que el proceso de programación ha sido en un entorno interactivo que ha facilitado el aprendizaje del lenguaje mencionado.

JupyterLab ha sido ejecutado dentro de Anaconda ya que incluye múltiples librerías útiles por defecto como Pandas y permite el uso y la gestión de entornos virtuales de manera sencilla mediante conda.

### 4.2 Preparación librerías

Una vez preparado el entorno, descargado Anaconda y abierto el primer notebook dentro de JupyterLab, se ha iniciado el proceso de desarrollo empleando las siguientes librerías, con sus funciones indicadas:

1. pandas: para el análisis de datos y su manipulación. Permite cargar, explorar y transformar los datos.

2. glob: para la búsqueda de archivos mediante patrones.
3. os: gestión de rutas y operaciones del sistema de archivos.
4. darts: librería especializada en modelado de series temporales, para trabajar con TimeSeries.
5. darts.models.ARIMA: modelo ARIMA para series temporales dentro de la librería Darts. Es una alternativa a statsmodels (que también se ha empleado).
6. requests: para solicitar API utilizando Python.
7. selenium: para automatizar la extracción de datos de páginas web.
8. pmdarima: para hacer AutoARIMA y elegir los valores óptimos para el modelo ARIMA
9. statsmodels: para construir manualmente modelos ARIMA y SARIMA.
10. skforecast: para usar modelos de scikit-learn para series temporales.
11. numpy: para cálculos numéricos.
12. matplotlib: para crear gráficos.
13. SHAP: para aplicar la técnica SHAP, tiene muchas funciones.
14. LIME: para aplicar la técnica LIME

## 5. Capítulo V: Preparación de datos

En este capítulo se detallará el método de investigación, extracción, preprocesamiento y primeras conclusiones de los datos necesarios para el desarrollo de modelos. La estructura del capítulo será por tipología de datos, ya que cada una requiere de fuente y tratamiento diferente.

### 5.1 Datos demanda energética

En primer lugar, se detallará el método de extracción de los datos históricos de la variable dependiente, los valores horarios de demanda eléctrica, que servirán para entrenar los modelos creados.

Para ello, es necesario enviar un correo electrónico a REE para recibir un token que permite hacer peticiones personalizadas y seguras a través de su API.

Prácticamente un día después de la solicitud del token, se pudo hacer uso de él. El primer paso ha sido programar en Python para automatizar la descarga de datos, dividiendo el periodo de 5 años seleccionados en periodos más pequeños de 4 meses, para evitar errores en la extracción de datos.

Una vez hecho, se vuelve a crear otro programa en Python para unificar los documentos cvs descargados en uno solo. De esta manera será más sencillo trabajar en un futuro con nuestros modelos. Es necesario para ello, además de unificar, limpiar los datos para eliminar cualquier fila donde pueda haber ceros y dejarlos ordenados para trabajar con ellos más adelante.

Por otro lado, REE proporciona los datos en formato UTC, ya que es el estándar utilizado globalmente, pero eso quiere decir que es necesario ajustar la hora al formato de la zona horaria a la que se ajusta España (CET). En este caso, es algo que se ha tenido en cuenta al final, por lo que se mostrará después de la extracción de datos de festividades.

Finalmente, se obtiene un solo archivo csv con dos columnas: en la primera, la fecha y hora y en la segunda, la demanda horaria en ese momento.

## 5.2 Datos factores climatológicos

Como se ha mencionado anteriormente, la temperatura es uno de los factores que más se espera que influya en el modelo, pero se busca exactitud en la previsión, por lo que también se incluirán otras variables como precipitación, viento y nubosidad. La idea es predecir la demanda de manera horaria, por lo que los datos se deben extraer del mismo modo.

Para que los datos climatológicos sean representativos de toda la península, sin trabajar con una cantidad tan grande de datos que entorpezca el proceso, se han seleccionado 10 estaciones o ubicaciones estratégicas:

- Madrid, Barcelona, Zaragoza, Bilbao, Valencia: zonas muy pobladas e industrializadas, donde existe una alta correlación entre clima y demanda eléctrica urbana, industrial y de servicios.
- Sevilla, Valladolid, A Coruña: representan extremos climáticos (Sevilla, calor; Valladolid, frío; A Coruña, alta pluviosidad y humedad), por lo que son útiles para modelar el impacto del clima en el uso de la climatización (calefacción/aire acondicionado).
- Málaga, Córdoba: clima mediterráneo con fuerte componente de estacionalidad. Es interesante incluir también comportamientos no lineales y picos estacionales en nuestro modelo para asegurar su viabilidad.

De esta manera, se tienen en cuenta los principales factores climatológicos que tienen más impacto en España para representar adecuadamente la diversidad climática y de consumo eléctrico.

Una vez detallada la selección, se procede a la descarga de los datos.

En una primera instancia, se intentó descargar los datos del servicio de OpenData de AEMET. Para ello, fue necesario presentar una solicitud formal, ya que los datos que se necesitan para este proyecto son datos climatológicos horarios (no diarios). El servicio gratuito de AEMET OpenData solo permite descargar valores climatológicos diarios, y se

deben descargar por estación meteorológica con un intervalo de tiempo máximo de 4 meses por descarga. Por lo tanto, se presentó una solicitud a través de la sede electrónica de AEMET indicando el fin educativo de este proyecto e indicando las variables solicitadas.

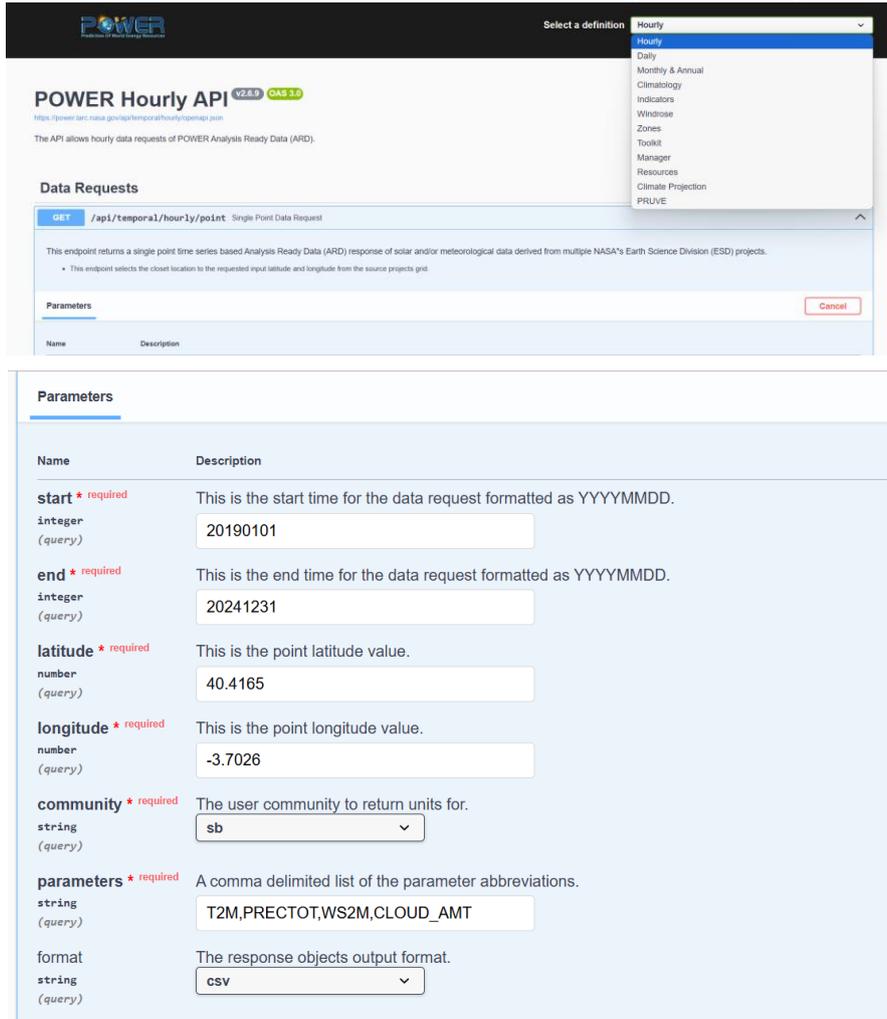
Sin embargo, este método no tuvo éxito ya que para acceder a los datos AEMET cobra unos 8000 euros.

Finalmente, se tuvo que buscar una alternativa, encontrando la web *NASA power*, que permite descargar datos horarios sin ningún tipo de trámite burocrático previo.

La única diferencia es que los valores climatológicos no se buscan por estación, sino que se solicitan indicando latitud y longitud de las ciudades seleccionadas. La interfaz es intuitiva y lo único que hay que tener en cuenta, además de lo mencionado, es la comunidad científica. Se ha marcado 'SB' por ser la estándar. Se puede observar a continuación en la Ilustración 8 la solicitud de datos para Madrid (longitud -3.7038, latitud 40.4168) en el intervalo de años con los que se trabajará.

Para encontrar la latitud y longitud de las ciudades indicadas anteriormente, se hace uso de la web geodatos.

Finalmente, la petición de datos a NASAPower se visualiza de la manera indicada en la Figura 11:



The screenshot shows the NASA POWER API web interface. At the top, there is a 'Select a definition' dropdown menu with 'Hourly' selected. Below this, the 'POWER Hourly API' is displayed with version information (V2.5.9, OAS 3.0). The main section is titled 'Data Requests' and shows a 'GET /api/temporal/hourly/point' endpoint. A description explains that this endpoint returns a single point time series based Analysis Ready Data (ARD) response of solar and/or meteorological data. Below the description is a 'Parameters' section with a 'Cancel' button. The parameters are listed in a table:

Name	Description
<b>start</b> * required integer (query)	This is the start time for the data request formatted as YYYYMMDD. 20190101
<b>end</b> * required integer (query)	This is the end time for the data request formatted as YYYYMMDD. 20241231
<b>latitude</b> * required number (query)	This is the point latitude value. 40.4165
<b>longitude</b> * required number (query)	This is the point longitude value. -3.7026
<b>community</b> * required string (query)	The user community to return units for. sb
<b>parameters</b> * required string (query)	A comma delimited list of the parameter abbreviations. T2M,PRECTOT,WS2M,CLOUD_AMT
<b>format</b> string (query)	The response objects output format. csv

Figura 11: Solicitud de datos web NASA POWER. Ejemplo Madrid

Una vez rellenos los campos, solo hay que descargarlos y ordenarlos en el archivo csv. Se repite el proceso con el resto de las ciudades y finalmente se unifican los valores a través de un nuevo código de Python.

Por simplicidad y precisión, se acaban empleando los datos proporcionados por el tutor Pablo, ya que la web NASAPower no recoge los datos a través de estaciones meteorológicas como AEMET, sino que recoge datos de temperaturas interpolando en puntos del territorio.

Estos datos proporcionados por el tutor están agrupados de forma horaria desde 2018 hasta 2022. Las temperaturas no se indican como una temperatura media sino como tamax y tamin, por lo que para realizar el modelo de manera cómoda se hace la media entre ambas para obtener una temperatura media.

De nuevo, es necesario hacer una limpieza de datos para eliminar aquellas filas en las que faltan datos. Puesto que el número de filas en las que faltan datos es tan pequeño en proporción al volumen total de datos manejados, la manera de hacerlo elegida es simplemente eliminando dichas filas.

Se podría optar por otras alternativas, como sustituyendo los datos por valores razonablemente similares a los del rango en el que se encuentran (haciendo una media o similar), pero la proporción es pequeña y por ende eliminar las filas es un mecanismo práctico y eficiente.

El archivo con los datos meteorológicos está dividido por horas y por ciudades como se muestra en la figura 12. Por ello, lo primero que se hace es agregar una nueva columna con una nueva variable `temp_media` haciendo la media entre `tamax` y `tamin`. Posteriormente, puesto que se tiene la demanda eléctrica nacional, se hace una media de temperaturas entre todas las ciudades que se incluyen en el set de datos. Así, se obtiene la temperatura media (nacional) por cada hora como muestra la figura 13.

	A	B	C	D	E	F	G	H	I	J	K	L
1	<code>datetime_utc</code>	<code>local_datetime</code>	<code>date_local_int</code>	hora	name	idema	id_municip	tamax	tamin	prec	vmax	inso
2	2019-01-01 00:00:00	2019-01-01 00:00:00	20190101	2	VALLADOLID	2422	47186	1.6	0.8	0.0	1.0	0.0
3	2019-01-01 01:00:00	2019-01-01 01:00:00	20190101	3	VALLADOLID	2422	47186	0.8	0.2	0.0	1.1	0.0
4	2019-01-01 02:00:00	2019-01-01 02:00:00	20190101	4	VALLADOLID	2422	47186	0.2	-0.4	0.0	1.1	0.0
5	2019-01-01 03:00:00	2019-01-01 03:00:00	20190101	5	VALLADOLID	2422	47186	-0.3	-0.8	0.0	1.1	0.0
6	2019-01-01 04:00:00	2019-01-01 04:00:00	20190101	6	VALLADOLID	2422	47186	-0.7	-1.2	0.0	1.1	0.0
7	2019-01-01 05:00:00	2019-01-01 05:00:00	20190101	7	VALLADOLID	2422	47186	-1.0	-1.4	0.0	0.9	0.0
8	2019-01-01 06:00:00	2019-01-01 06:00:00	20190101	8	VALLADOLID	2422	47186	-1.4	-1.8	0.0	0.9	0.0
9	2019-01-01 07:00:00	2019-01-01 07:00:00	20190101	9	VALLADOLID	2422	47186	-1.8	-1.9	0.0	0.9	0.0
10	2019-01-01 08:00:00	2019-01-01 08:00:00	20190101	10	VALLADOLID	2422	47186	-1.9	-2.4	0.0		0.0

Figura 12: Vista del archivo de datos meteorológicos (elaboración propia)

	<code>datetime2_local</code>	<code>temp_media</code>
0	2019-01-01 01:00:00+01:00	5.007692
1	2019-01-01 02:00:00+01:00	4.480769
2	2019-01-01 03:00:00+01:00	4.094231
3	2019-01-01 04:00:00+01:00	3.751923
4	2019-01-01 05:00:00+01:00	3.440385

Figura 13: Vista de columnas hora y `temp_media`

A continuación, se agrega la nueva variable `temp_media` al archivo en el que se tiene la demanda eléctrica y la variable `fest` por cada hora, quedando de la siguiente manera:

### 5.3 Datos festividades

La última variable dependiente que se tendrá en cuenta en el modelo será la de festividad. Esta variable tiene gran impacto en la demanda, como se ha mencionado, pues los hábitos de las personas cambian radicalmente en días festivos.

Para incluir estos datos en el modelo, se emplea Python de nuevo. En este caso, se realiza directamente en el mismo archivo en el que se tiene la demanda eléctrica horaria junto a una columna de fecha y hora correspondientes denominada 'datetime2'. Para la variable festividad, se realiza en dos pasos. Primero se introduce la columna 'type' en el archivo que, basándose en la fecha (ya en formato yyyy-mm-dd) asigna un valor del 1 al 4 dependiendo del día de la semana:

1 para los días laborables de lunes a jueves, que se puede considerar que presentan un comportamiento similar

2 para viernes

3 para sábado

4 para domingos y festivos

Además, se agregan de manera manual los festivos nacionales como el 1 de enero o 6 de enero. De esta manera, la visión de esta nueva variable sería la mostrada en la Figura 14:

	<b>datetime2</b>	<b>value</b>	<b>type</b>
<b>0</b>	2019-01-01 00:00:00	23459.000000	1
<b>1</b>	2019-01-01 01:00:00	22781.000000	1
<b>2</b>	2019-01-01 02:00:00	21448.500000	1
<b>3</b>	2019-01-01 03:00:00	20262.166667	1
<b>4</b>	2019-01-01 04:00:00	19463.500000	1
...	...	...	...
<b>8779</b>	2024-12-31 19:00:00	31136.333333	1
<b>8780</b>	2024-12-31 20:00:00	30291.666667	1
<b>8781</b>	2024-12-31 21:00:00	28500.250000	1
<b>8782</b>	2024-12-31 22:00:00	25583.500000	1
<b>8783</b>	2024-12-31 23:00:00	24053.416667	1

Figura 14: Columna type para clasificación de los días de la semana

Posteriormente, se agrega una variable adicional: la variable 'fest', que indica si el día indicado por 'datetime2' es festivo o no. Como se ha adelantado anteriormente, se tratará a los domingos como festivos por presentar un comportamiento similar, por lo que la columna de la variable 'fest' asigna un '0' a aquellos días que no son ni festivos ni domingo y un '1' para domingos y festivos. A continuación, en la Figura 15, se presenta la visión de la nueva variable 'fest'. Se puede apreciar el 1 asignado al primer día del año 2019, 1 de enero, festivo.

	<b>datetime2</b>	<b>value</b>	<b>type</b>	<b>fest</b>
<b>0</b>	2019-01-01 00:00:00	23459.000000	1	1
<b>1</b>	2019-01-01 01:00:00	22781.000000	1	1
<b>2</b>	2019-01-01 02:00:00	21448.500000	1	1
<b>3</b>	2019-01-01 03:00:00	20262.166667	1	1
<b>4</b>	2019-01-01 04:00:00	19463.500000	1	1
...	...	...	...	...

Figura 15: columna fest para asignación de festividad

Finalmente, como se ha indicado durante el proceso de tratamiento de los datos de demanda, es necesario modificar la hora para ajustarla a la zona horaria de Europa Central (CET), por lo que se hace una vez unificado la demanda con la festividad, quedando de la siguiente manera con la nueva variable datetime2\_local:

	datetime2	datetime2_local
0	2019-01-01 00:00:00	2019-01-01 01:00:00+01:00
1	2019-01-01 01:00:00	2019-01-01 02:00:00+01:00
2	2019-01-01 02:00:00	2019-01-01 03:00:00+01:00
3	2019-01-01 03:00:00	2019-01-01 04:00:00+01:00
4	2019-01-01 04:00:00	2019-01-01 05:00:00+01:00

Figura 16: conversión a formato horario CET

En la figura 17 se puede observar la vista del archivo con las columnas de las nuevas variables agrupadas.

	datetime2_local	value	fest	temp_media
0	2019-01-01 01:00:00+01:00	23459.000000	1	5.007692
1	2019-01-01 02:00:00+01:00	22781.000000	1	4.480769
2	2019-01-01 03:00:00+01:00	21448.500000	1	4.094231
3	2019-01-01 04:00:00+01:00	20262.166667	1	3.751923
4	2019-01-01 05:00:00+01:00	19463.500000	1	3.440385

Figura 17: Vista de columnas hora, demanda, festividad y temp\_media

Debido a que la variable 'fest' es dicotómica (1 o 0), surge un pequeño inconveniente al tener en cuenta las festividades por provincia. Tenemos la demanda eléctrica nacional, pero el hecho de que sea festivo en la Comunidad de Madrid, por ejemplo, influye directamente en la demanda eléctrica nacional de ese día, pero no de la misma manera que si fuera un festivo nacional. Por ende, se tienen dos alternativas para contabilizar este tipo de sucesos:

Por un lado, se podría crear una variable fest\_ponderada, que en lugar de ser 1/0, represente el porcentaje de población de la provincia donde es festivo ese día. Por ejemplo, si el 30% de la población nacional tiene un día festivo, la variable fest\_ponderada vale 0,3.

Como enfoque más simple y pragmático, se tendrá en cuenta solo aquellas comunidades grandes como Madrid, Cataluña y Andalucía, las comunidades autónomas con mayor población. Dado que estas regiones concentran una parte muy significativa del consumo eléctrico nacional, sus patrones festivos tendrán un impacto más acusado en la demanda global. Aunque es un método menos preciso, reduce complejidad al modelo, y aporta agilidad al proceso.

A continuación, en la figura 15 se puede observar el resultado con dos días de prueba: el día del trabajador (1 de mayo) y el día de Andalucía (28 de febrero), ambos marcados como 'fest'=1, siguiendo el planteamiento expuesto. Por otro lado, en la figura 16 se observa, para comprobar, un día cualquiera como el 1 de abril, marcado con fest=0.

	datetime2_local	value	temp_media	fest		datetime2_local	value	temp_media	fest
<b>1391</b>	2019-02-28 00:00:00+01:00	28550.000000	10.564583	1	<b>2877</b>	2019-05-01 00:00:00+02:00	28246.500000	16.230769	1
<b>1392</b>	2019-02-28 01:00:00+01:00	26742.166667	9.532000	1	<b>2878</b>	2019-05-01 01:00:00+02:00	26061.333333	15.451923	1
<b>1393</b>	2019-02-28 02:00:00+01:00	25207.500000	8.964583	1	<b>2879</b>	2019-05-01 02:00:00+02:00	24449.500000	14.836538	1
<b>1394</b>	2019-02-28 03:00:00+01:00	24037.333333	8.400000	1	<b>2880</b>	2019-05-01 03:00:00+02:00	22962.333333	13.829167	1
<b>1395</b>	2019-02-28 04:00:00+01:00	23705.833333	7.850000	1	<b>2881</b>	2019-05-01 04:00:00+02:00	21648.500000	13.253846	1
...	...	...	...	...	...	...	...	...	...
<b>27622</b>	2022-02-28 19:00:00+01:00	28429.333333	15.842308	1	<b>29107</b>	2022-05-01 19:00:00+02:00	20439.666667	22.323077	1
<b>27623</b>	2022-02-28 20:00:00+01:00	30888.166667	14.173077	1	<b>29108</b>	2022-05-01 20:00:00+02:00	20794.666667	21.488000	1
<b>27624</b>	2022-02-28 21:00:00+01:00	32468.333333	12.848000	1	<b>29109</b>	2022-05-01 21:00:00+02:00	21227.000000	19.892308	1
<b>27625</b>	2022-02-28 22:00:00+01:00	31955.000000	11.798077	1	<b>29110</b>	2022-05-01 22:00:00+02:00	22223.166667	18.373077	1
<b>27626</b>	2022-02-28 23:00:00+01:00	29361.166667	10.822000	1	<b>29111</b>	2022-05-01 23:00:00+02:00	23971.666667	17.151923	1

Figura 18: Vista final con días festivos

	datetime2_local	value	temp_media	fest
<b>2157</b>	2019-04-01 00:00:00+02:00	28610.666667	12.084615	0
<b>2158</b>	2019-04-01 01:00:00+02:00	26701.833333	11.836538	0
<b>2159</b>	2019-04-01 02:00:00+02:00	24549.500000	11.498077	0
<b>2160</b>	2019-04-01 03:00:00+02:00	22938.500000	11.169231	0
<b>2161</b>	2019-04-01 04:00:00+02:00	21819.666667	11.038462	0
...	...	...	...	...
<b>28387</b>	2022-04-01 19:00:00+02:00	28471.833333	12.682000	0
<b>28388</b>	2022-04-01 20:00:00+02:00	28209.000000	11.944000	0
<b>28389</b>	2022-04-01 21:00:00+02:00	28536.333333	10.308000	0
<b>28390</b>	2022-04-01 22:00:00+02:00	29922.500000	8.968000	0
<b>28391</b>	2022-04-01 23:00:00+02:00	31789.666667	8.120000	0

Figura 19: Vista final dataset día 1 de abril

## 6. Capítulo VI: Modelado

A continuación, se detallará el método de modelado paso por paso.

Antes de proceder al modelado, se divide el set de datos en dos partes: una reservada para el entrenamiento de los modelos y otra para su validación. Este paso es importante y no debe faltar ya que permite evaluar la capacidad de predicción del modelo con datos que todavía no ha visto. De este modo se evita el sobreajuste y se obtienen modelos generalizables.

En este proyecto, la intención es destinar el 80% del set de datos al entrenamiento y el 20% restante a la validación del modelo, una división común en modelado de series temporales. Es una buena proporción ya que permite disponer de un volumen suficientemente grande para que el modelo 'aprenda', mientras que también se reserva una parte suficientemente representativa para poder evaluar las predicciones hechas.

Sin embargo, no será una división aleatoria sino una división por fecha. Esto se debe a que la variable dependiente (la demanda) depende fuertemente del tiempo en el que se mida, pues como se ha estudiado existen tendencias, estacionalidades y dependencia temporal en las observaciones. Si se mezclase de manera aleatoria, se estarían mezclando datos pasados y futuros sin respetar el orden temporal y ello llevaría a predicciones incorrectas. Esto lleva a que la proporción ideal inicial varíe un poco, ya que los datos presentados abarcan desde 2019 hasta 2022. La división realizada es la siguiente:

- Entrenamiento: 01-01-2019 hasta 31-12-2021, 26.211 muestras
- Validación: 01-01-2022 hasta 31-12-2022, 8.531 muestras

Cabe señalar que, en la limpieza de datos, se eliminaron las filas cuyos campos no estaban rellenos. Por ello, hay menos observaciones que horas en un año.

Finalmente, por tanto, se tienen 34.742 observaciones, y la proporción entrenamiento-validación resulta ser 75,4% - 24,6%, cercana a la que se buscaba idealmente.

## 6.1 Modelos

Antes de comenzar a modelar, unos detalles respecto de la división en dos sets: entrenamiento y validación.

En primer lugar, se procede a realizar una prueba de Dickey-Fuller (ADF) para comprobar la estacionariedad de la serie. La hipótesis nula es que la serie no es estacionaria, por lo que si el pvalor de la prueba resulta menor que los valores habituales (0,05), se rechaza la hipótesis nula y por ende, la serie es estacionaria.

---

```
ADF Statistic: -16.0202
p-value:      0.0000
Critical Values:
  1%: -3.4305
  5%: -2.8616
 10%: -2.5668
```

---

*Figura 20: Resultados prueba ADF*

Tras la realización de la prueba, se observa un p valor de aproximadamente 0, por lo que se rechaza la hipótesis nula. A pesar de las tendencias observadas y expuestas en el análisis inicial de demanda eléctrica inicial, al realizar este test, la conclusión resulta diferente: la serie es estacionaria. Sin embargo, puede ser debido a que el espectro temporal no captura las tendencias históricas de la demanda eléctrica, ya que solo se están teniendo en cuenta los años 2019-2022.

### 6.1.1 ARIMA

En primer lugar, se ha optado por utilizar el modelo de series temporales ARIMA como primer método para predecir la demanda. Para ello, se ha utilizado la librería `darts` en Python, que permite trabajar con series temporales, con el apoyo de la web *unit8*, citada en más detalle en el apartado de referencias.

El tipo de dato en `darts` es `TimeSeries`, por lo que se cargan en este formato las variables a emplear.

Como se ha mencionado en el Capítulo III, el modelo ARIMA es también aplicable en series de datos que no son estacionarias, es decir, con cierta tendencia, por lo que no se requiere comprobar estacionariedad y el parámetro  $d$  se encarga de, a través de la diferenciación, acabar haciéndola estacionaria.

A continuación, se detalla el siguiente paso, la elección de los parámetros  $p$ ,  $q$  y  $d$  del modelo.

Como se ha mencionado en un principio, se buscan valores prácticos. Una manera de seleccionar estos parámetros es utilizando la función AutoARIMA, como propone la web mencionada. Para ello, se utiliza la librería `pmdarima`. Es conveniente hacerlo de esta manera ya que la función selecciona automáticamente los valores de  $p$ ,  $d$  y  $q$ , ajustando el modelo de manera eficiente.

Se debe tener en cuenta que el modelo ARIMA es univariante, lo que implica que solo se basa en la propia serie de demanda para predecir valores futuros. Si se quisiera tener en cuenta el resto de variables, se procedería con un modelo ARIMAX, cuya formulación es muy parecida, pero admite un vector de variables exógenas.

Una vez ejecutada la acción `auto_arima` de `pmdarima` con el set de entrenamiento, tras unos 7 minutos, se ha obtenido el mejor modelo, siendo: `ARIMA(2,1,5)(0,0,0)[0]`

Se comprueba que la serie no era estacionaria, puesto que el mejor modelo sí presenta diferenciación según esta función. La prueba ADF puede haber tenido ese resultado por el horizonte temporal (de solo cuatro años) con el que se trabaja, más adelante se vuelve a realizar tras una diferenciación para comprobar.

La función `auto_arima` selecciona este modelo tras evaluar los AIC de todas las opciones, escogiendo el de mínimo AIC, en este caso  $AIC = 419.140.027$ , tal y como se observa en la figura 21. Por otro lado, en la figura 22 se puede comprobar que, para el modelo escogido, todos los coeficientes son significativos ( $p$  valor aprox. 0)

Si se observan los parámetros seleccionados, se tiene:  $p=2$ ,  $d=1$  y  $q=5$ . Los parámetros  $p=2$  y  $q=5$  capturan la dependencia autorregresiva y de media móvil, mientras que  $d=1$  corresponde a la primera diferencia necesaria para garantizar la estacionariedad.

```

=====
No. Observations:          26211
Log Likelihood             -209562.014
AIC                       419140.027
BIC                       419205.418
HQIC                      419161.144
  
```

Figura 21: Resultado AIC modelo ARIMA seleccionado

```

=====
              coef      std err          z      P>|z|      [0.025      0.975]
-----+-----
ar.L1          1.7098         0.001    1178.716     0.000         1.707         1.713
ar.L2         -0.9803         0.001   -694.176     0.000        -0.983        -0.978
ma.L1         -0.7388         0.005   -144.343     0.000        -0.749        -0.729
ma.L2         -0.1907         0.007    -28.392     0.000        -0.204        -0.178
ma.L3          0.3759         0.007     51.496     0.000         0.362         0.390
ma.L4          0.2079         0.008     25.198     0.000         0.192         0.224
ma.L5          0.0635         0.007     9.567      0.000         0.050         0.077
sigma2       5.085e+05     2983.549    170.451     0.000       5.03e+05       5.14e+05
=====
Ljung-Box (L1) (Q):                0.06   Jarque-Bera (JB):                20790.03
Prob(Q):                            0.81   Prob(JB):                          0.00
Heteroskedasticity (H):              0.97   Skew:                               -0.62
Prob(H) (two-sided):                 0.12   Kurtosis:                           7.18
=====
  
```

Figura 22: Modelo ARIMA seleccionado

El análisis de Ljung-Box aplicado a los residuos resulta en un valor de  $p=0.81$ . Al ser muy superior al umbral habitual, no se puede rechazar la hipótesis nula (ausencia de correlación) y por ende, no hay autocorrelación en los residuos, tal y como se quería testar.

### 6.1.2 SARIMA

El modelo SARIMA, como se ha expuesto en capítulos anteriores es una variación del modelo ARIMA con la incorporación del componente estacional. Al introducir este componente a través de la variable  $s$ , se le debe asignar también un valor.

Al agregar esta variable, el coste computacional de este modelo aumenta considerablemente y en consecuencia salta un error de memoria: *MemoryError*.

El problema reside en la cantidad de iteraciones que debe realizar el programa, intentando ajustar muchos  $(p,d,q)$   $(P, D, Q)$ . El modelo SARIMA monta matrices de órdenes demasiado elevados que, en la memoria del dispositivo en el que se programa, no caben.

Para solventar este error, en lugar de ejecutar este exhaustivo proceso, se procede a fijar un valor de parámetros SARIMA inicial basada en criterios teóricos. A partir de ahí, se van ajustando manualmente los parámetros en sucesivas iteraciones. Cada combinación, se evalúa mediante su AIC y se recoge en la tabla 3, lo que permite encontrar un modelo eficiente evitando el alto coste computacional de probar todas las configuraciones automáticamente.

Para escoger inicialmente los parámetros, se procede a realizar un análisis de la serie.

En primer lugar, un análisis para identificar la estacionalidad. Para ello, se separa la serie en sus componentes: tendencia, estacionalidad y residuos. De esta manera, se consigue observar la periodicidad de manera clara, aislándola del ruido o la tendencia.

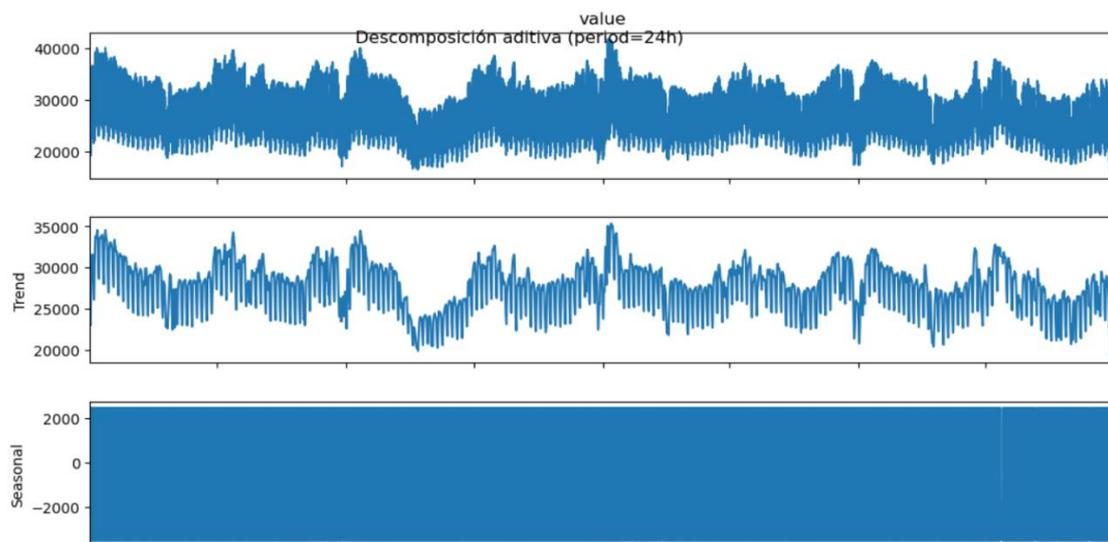
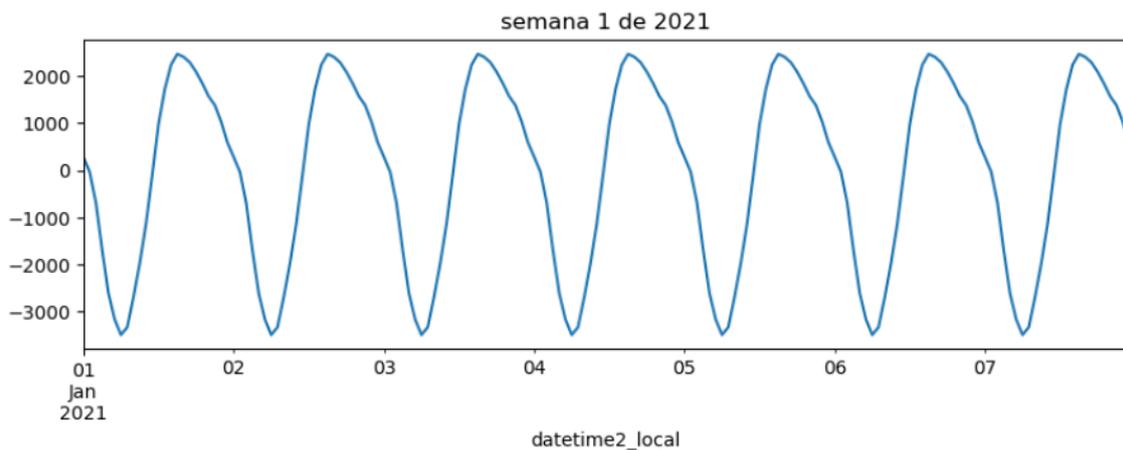


Figura 23: Serie completa y debajo, las componentes de tendencia y estacionalidad

Lo que se busca es comprobar el componente estacionalidad, pero en un rango tan amplio no se logra ver. Para ello, se realiza una ampliación en, por ejemplo, la primera semana de 2021. Se observa la serie con el componente de estacionalidad aislado en la figura 24 y se confirma visualmente un patrón que se repite cada 24h.



*Figura 24: Ampliación componente estacional en semana 1 de 2021.*

Por ende, el valor de  $s$  en el modelo SARIMA será  $s=24$ . Los datos son horarios y si se quiere representar un patrón diario, la periodicidad es 24 horas.

A continuación, se busca asignar valores al resto de variables:  $p$ ,  $d$ ,  $q$  y  $P$ ,  $D$ ,  $Q$ .

Una vez se tiene dividida la serie en componentes, es más sencillo extraer la  $D$ , diferenciación estacional, diferenciando la componente estacional y observando si con una diferenciación es suficiente para ver la serie plana.

Se vuelve a ejecutar el test ADF de dos formas, primero tras una diferenciación estacional y después tras una diferenciación normal. Resultando ambas pruebas en un  $p$  valor de 0, por lo que se da por válido los parámetros  $D=1$  y  $d=1$ .

```
ADF statistic (seasonal diff): -26.5564
p-value                       : 0.0000
```

*Figura 25: Resultado segunda prueba ADF después de diff estacional*

En la figura 26 se incluyen visualmente dos pruebas: ACF y PACF (función de autocorrelación y función de autocorrelación parcial, respectivamente). La ACF incluye

todas las correlaciones, mientras que la PACF solo la correlación directa. Con estas dos gráficas, se pueden identificar los valores razonables de  $p$  y  $q$ .

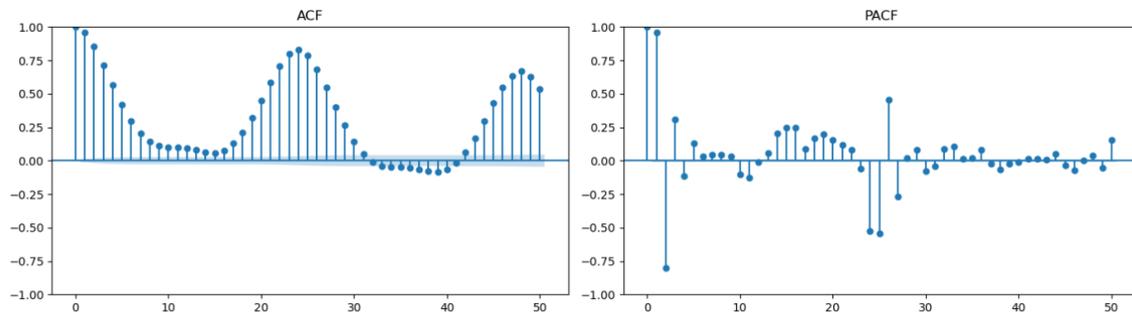


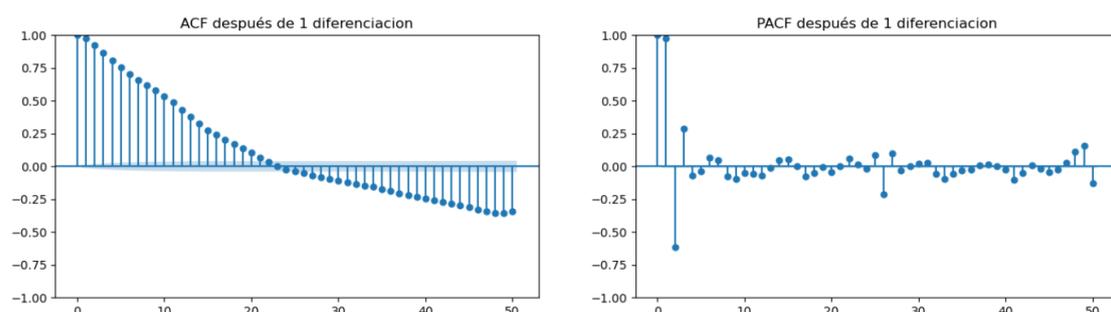
Figura 26: ACF y PACF

Observando la gráfica ACF se puede identificar el valor del parámetro  $q$ , ya que expresa la relación entre un punto y sus valores pasados. Este gráfico cae muy despacio y muestra correlaciones significativas hasta que se va acercando a cero (tras unas 5 o 6 barras). Ello sugiere un valor de un componente de media móvil bastante elevado, por lo que se puede empezar por  $q=5$  e ir bajando para comprobar.

Por otro lado, el gráfico PACF es útil para establecer el parámetro  $p$ , ya que elimina los efectos de rezago. En este caso solo se observa un pico fuerte y a partir del segundo elemento, ya está casi todo por debajo del umbral, por lo que se puede interpretar un valor de  $p=1$ .

Además, esto indica también que la serie presenta tendencia, por lo que el valor asignado de  $d=1$  es correcto.

En la figura 27 se observan las mismas pruebas, pero, este caso, se han realizado después de la diferenciación, con el objetivo de identificar los parámetros  $P$  y  $Q$ .



*Figura 27: ACF y PACF tras 1 diferenciación (s=24h)*

En el caso del gráfico PACF es muy similar, y sugiere  $P=1$ . El de ACF presenta un pico grande que disminuye paulatinamente. Se puede interpretar  $Q=1$ .

Ahora que ya se han escogido los parámetros, se procede a entrenar el modelo inicial: SARIMA(1,1,5)(1,1,1)[24]. Posteriormente, se evalúan dos modelos reduciendo el parámetro  $q$  y se recogen en la tabla 3, junto a sus respectivos valores de AIC.

Modelo	AIC
<b>SARIMA(1,1,5)(1,1,1)[24]</b>	380.882,837
<b>SARIMA(1,1,3)(1,1,1)[24]</b>	379.163,291
<b>SARIMA(1,1,2)(1,1,1)[24]</b>	380.869,249

*Tabla 3: Comparación modelos SARIMA*

Por ende, de entre los modelos SARIMA analizados, el mejor modelo es SARIMA(1,1,3)(1,1,1)[24], puesto que es el de menor AIC. Cabe destacar el tiempo requerido para estas tres iteraciones, aproximadamente 15 minutos cada una. En la figura 28 se pueden observar los resultados del modelo seleccionado.

```

=====
SARIMAX Results
=====
Dep. Variable:                value    No. Observations:          26303
Model:                        SARIMAX(1, 1, 3)x(1, 1, [1], 24)  Log Likelihood             -189574.645
Date:                          Sat, 21 Jun 2025    AIC                       379163.291
Time:                          13:39:25      BIC                       379220.519
Sample:                        01-01-2019    HQIC                      379181.770
                                - 12-31-2021

Covariance Type:              approx
=====
              coef    std err          z      P>|z|      [0.025    0.975]
-----
ar.L1          0.9488     0.002    412.487    0.000     0.944     0.953
ma.L1         -0.1399     0.006   -22.200    0.000    -0.152    -0.128
ma.L2         -0.4849     0.006   -75.858    0.000    -0.497    -0.472
ma.L3         -0.3751     0.007   -55.788    0.000    -0.388    -0.362
ar.S.L24       0.3242     0.008    39.605    0.000     0.308     0.340
ma.S.L24      -0.8471     0.004  -217.146    0.000    -0.855    -0.839
sigma2        1.47e+05    1798.780    81.710    0.000    1.43e+05    1.51e+05
=====
Ljung-Box (L1) (Q):           10.68    Jarque-Bera (JB):          18501.57
Prob(Q):                      0.00     Prob(JB):                   0.00
Heteroskedasticity (H):       0.92     Skew:                       -0.07
Prob(H) (two-sided):          0.00     Kurtosis:                    7.11
=====

```

Figura 28: Modelo SARIMA (1,1,3)(1,1,1)[24]

### 6.1.3 XGBoost

Por último, se procede a evaluar el modelo XGBoost, que destaca por su rapidez y eficacia, como se ha presentado en el Capítulo III.

A diferencia de los modelos anteriores, este modelo permite la inclusión de variables exógenas de la base de datos, lo que contribuye a crear modelos más precisos.

Estas variables adicionales corresponden a las mencionadas y expuestas en la figura 19 y se presentan nuevamente en la figura 29 a modo de recordatorio, mostrando una vista general del dataset inicial.

	<b>datetime2_local</b>	<b>value</b>	<b>temp_media</b>	<b>fest</b>
<b>2157</b>	2019-04-01 00:00:00+02:00	28610.666667	12.084615	0
<b>2158</b>	2019-04-01 01:00:00+02:00	26701.833333	11.836538	0
<b>2159</b>	2019-04-01 02:00:00+02:00	24549.500000	11.498077	0
<b>2160</b>	2019-04-01 03:00:00+02:00	22938.500000	11.169231	0
<b>2161</b>	2019-04-01 04:00:00+02:00	21819.666667	11.038462	0
...	...	...	...	...
<b>28387</b>	2022-04-01 19:00:00+02:00	28471.833333	12.682000	0
<b>28388</b>	2022-04-01 20:00:00+02:00	28209.000000	11.944000	0
<b>28389</b>	2022-04-01 21:00:00+02:00	28536.333333	10.308000	0
<b>28390</b>	2022-04-01 22:00:00+02:00	29922.500000	8.968000	0
<b>28391</b>	2022-04-01 23:00:00+02:00	31789.666667	8.120000	0

*Figura 29: Vista del dataset (duplicado)*

Antes de entrenar el modelo, es importante preparar los datos de la manera adecuada. En este caso, las variables se codifican en formato DMatrix, estructura optimizada que XGBoost utiliza para mejorar rendimiento. A continuación, se definen los parámetros de entrenamiento, siguiendo valores típicos en modelos de predicción con este algoritmo:

- `n_estimators=100`. Este parámetro es el número de árboles que se construyen con el modelo. El valor 100 es una buena selección ya que no hará que el modelo tarde demasiado en el entrenamiento, pero es suficientemente alto como para ser preciso.
- `max_depth=5`. Es la profundidad máxima de cada árbol. Una profundidad máxima de 5 limita el sobreajuste del modelo.
- `learning_rate=0,1`. Es la tasa de aprendizaje, determina cuánto corregirá cada árbol nuevo.

Otro paso previo al entrenamiento del modelo XGBoost es definir el horizonte de información disponible para cada predicción. Es decir, hasta qué momento la serie puede usar datos pasados de la demanda para predecir. En este caso, para predecir la demanda en un instante de tiempo  $t$ , el modelo tiene disponibles los datos hasta el momento  $t-1$ .

De esta manera, XGBoost actúa como un modelo autorregresivo no lineal, ya que se entrena a partir de las observaciones pasadas para predecir las futuras.

Si se quisiera realizar una predicción de un día completo, por ejemplo, el modelo solo tendría acceso a los datos disponibles hasta una hora antes del cierre del día anterior. Si se quisiera predecir la demanda del 24 de junio de 2025, el modelo puede utilizar los datos disponibles hasta el 23 de junio de 2025 a las 23:00. Esto implica que cada una de las 24 predicciones horarias se realiza de forma secuencial o bien utilizando únicamente los rezagos permitidos generados a partir de los días anteriores.

Este control del horizonte informativo es esencial para asegurar que las métricas de validación (MAE y RMSE) obtenidas reflejan de forma realista el rendimiento del modelo en condiciones de despliegue.

Se comienza entrenando el modelo sin incluir variables exógenas, mostrando el resultado en la figura 30.

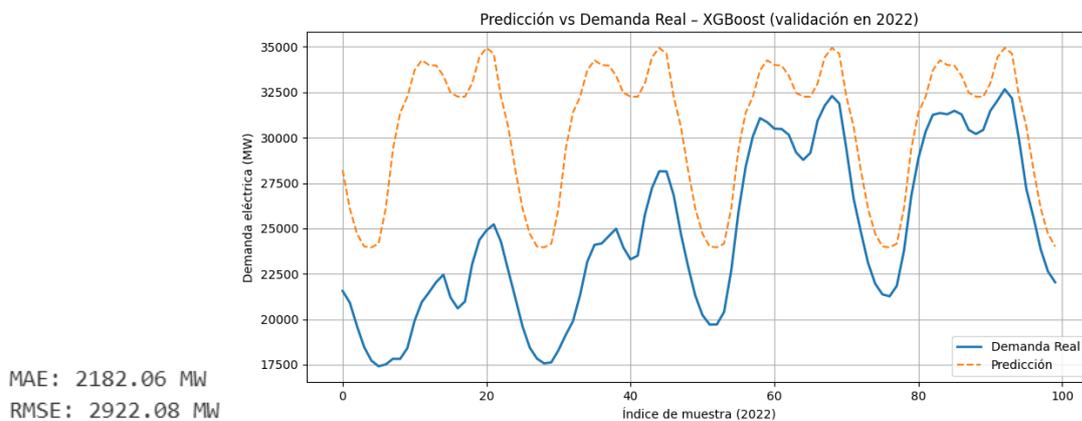


Figura 30: Modelo XGBoost sin variables exógenas

Posteriormente, se introduce la variable exógena fest (indicador de día festivo), generando una mejora que se puede percibir tanto visualmente como por las medidas de MAE y RMSE, como se observa en la figura 31.

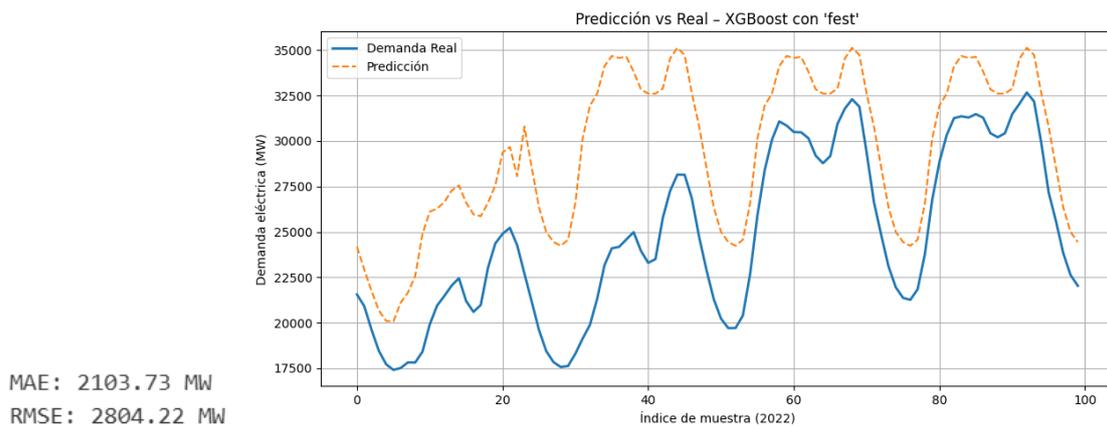


Figura 31: Modelo XGBoost con variable exógena fest

El modelo ha mejorado al agregar dicha variable exógena, pero sigue sin ser del todo preciso. A continuación, se agrega la siguiente variable exógena temp\_media, que mejora aún más la precisión del modelo, tal y como se observa en la figura 32.

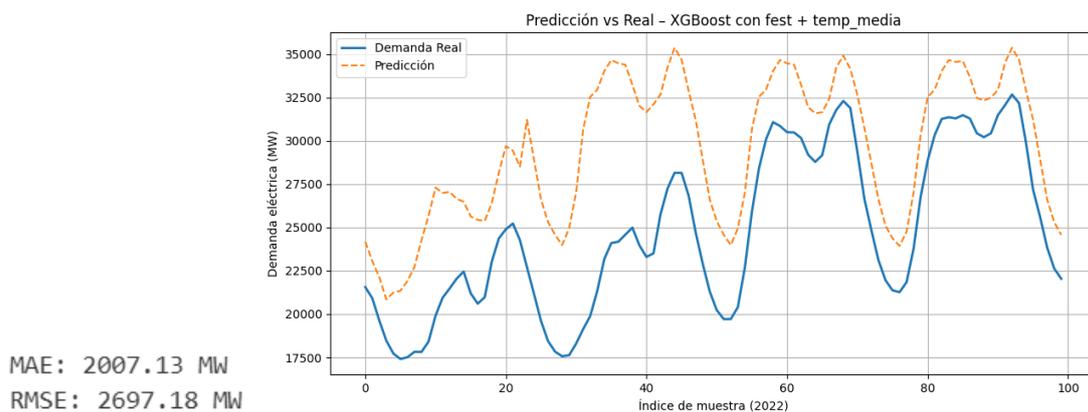


Figura 32: Modelo XGBoost con variables exógenas

Para seguir mejorando la precisión del modelo, se incorpora ahora una variables de retardo. En concreto, se agrega la variable value\_t-1, correspondiente a la demanda del día anterior a la misma hora. El resultado, visualizado en la figura 33, muestra una mejora significativa respecto del anterior: la forma de la curva predicha se asemeja notablemente a la real, lo que indica que el modelo ha captado los patrones horarios y diarios del consumo, aunque todavía se puede ajustar mejor.

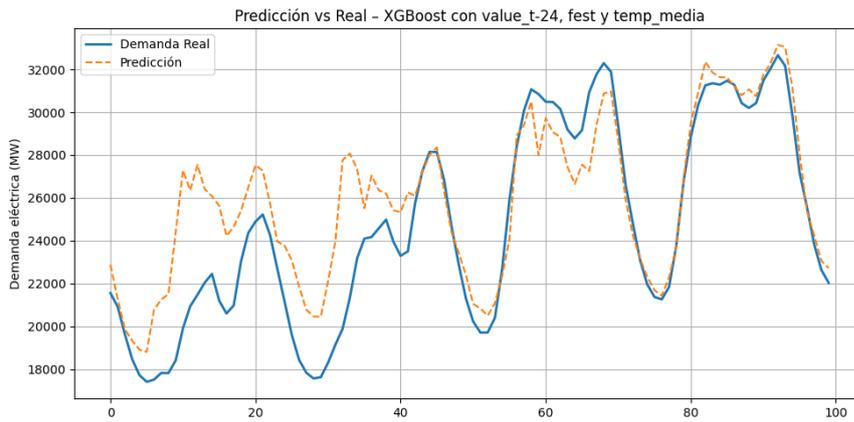


Figura 33: Modelo XGBoost con variables exógenas y variable de retardo value\_t-24

```
[Entrenamiento] MAE: 1438.50 MW | RMSE: 1924.35 MW
[Validación]   MAE: 1601.83 MW | RMSE: 2182.05 MW
```

Figura 34: Métricas modelo XGBoost Entrenamiento vs. Validación

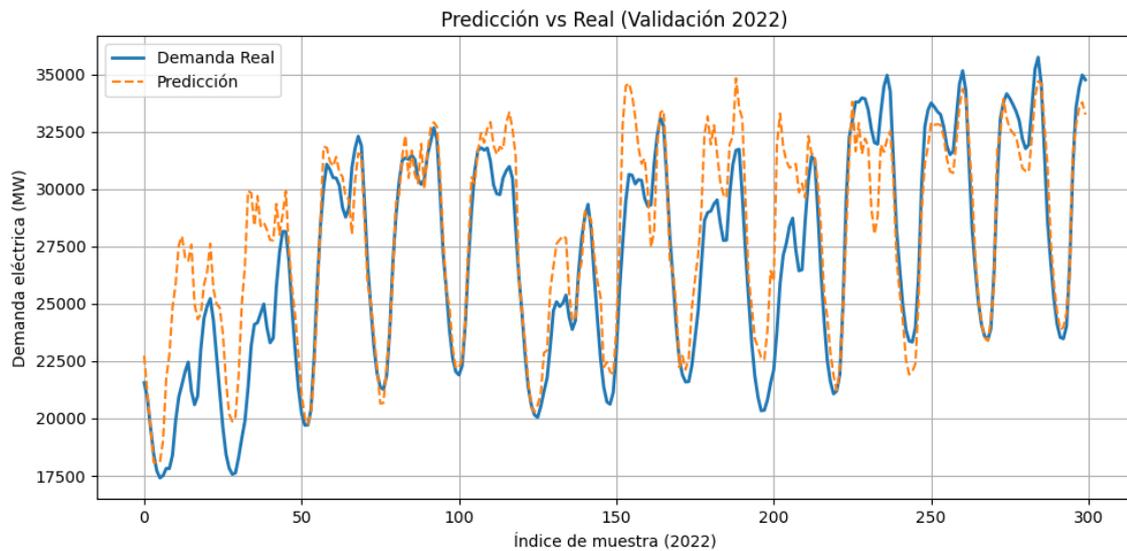
Cabe destacar que, en todos los experimentos, el tiempo de computación ha sido de apenas unas milésimas de segundo, confirmando la alta eficiencia computacional de XGBoost.

Como paso final, se propone el uso de una herramienta llamada GridSearch, que mejora el rendimiento y precisión del modelo XGBoost gracias a que busca los mejores parámetros para la serie de manera automática. Se usa GridSearchCV de scikit-learn para probar distintas combinaciones del número de árboles, la profundidad y la velocidad de aprendizaje

Finalmente, los parámetros seleccionados por GridSearch son:

- n\_estimators=300
- max\_depth=7
- learning\_rate=0,05

Cuando se entrena el modelo con dichos parámetros, se logra alcanzar la precisión mostrada en la figura 36, con mínimos MAE y RMSE.



*Figura 35: Modelo tras GridSearch*

[Entrenamiento] MAE: 1225.78 MW | RMSE: 1614.34 MW  
 [Validación] MAE: 1586.25 MW | RMSE: 2188.82 MW

*Figura 36: MAE y RMSE de Entrenamiento vs. Validación XGBoost*

Cuando los errores MAE y RMSE del set de validación son considerablemente mayores a los del set de entrenamiento, hay sobreajuste. En este caso, los errores del set de validación son aproximadamente un 30% mayor que los del set de entrenamiento, por lo que se puede considerar que existe sobreajuste. Para solucionarlo, se prueba un modelo más sencillo, disminuyendo `max_depth` y `n_estimators`.

Los nuevos parámetros serán:

- `n_estimators=100`
- `max_depth=5`
- `learning_rate=0,05`

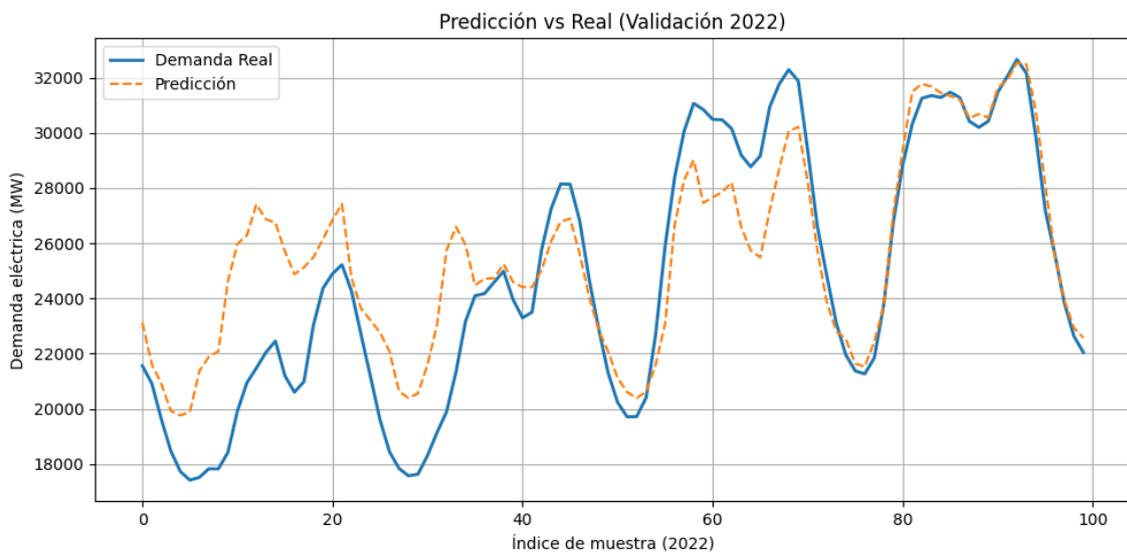


Figura 37: Modelo simplificado XGBoost

[Entrenamiento] MAE: 1497.65 MW | RMSE: 2040.36 MW  
 [Validación] MAE: 1626.85 MW | RMSE: 2210.45 MW

Figura 38: MAE y RMSE del modelo simplificado XGBoost

Se observa que ya no hay sobreajuste, pero este último modelo no presenta mejoras considerables y los resultados son muy similares a los de la figura 34.

## 7. Capítulo VII: Resultados

Esta fase de resultados es fundamental, tanto para analizar los resultados expuestos anteriormente, como para preparar la siguiente fase de análisis de explicabilidad con SHAP y LIME.

En este apartado, se presentan de forma sistemática los resultados obtenidos tras el entrenamiento y validación de los modelos en el apartado anterior, para poder compararlos y extraer conclusiones en el siguiente capítulo.

Para realizar las predicciones, se define el horizonte de información: el modelo solo tiene acceso a los valores de la variable demanda (value en el dataset) hasta el instante inmediatamente anterior, es decir, una hora antes del momento que se quiere predecir.

En otras palabras, al generar la previsión de todo el día 2025-06-23, se asume que el último dato real disponible de demanda es el de 2025-06-22 a las 23:00, y a partir de ahí, el modelo debe predecir en bloque las siguientes 24 horas. Esta restricción se mantiene tanto en los modelos estadísticos como en los modelos de machine learning, replicando así un contexto real de predicción operativa sin acceso a información futura.

Cada modelo: ARIMA, SARIMA y XGBoost se ha evaluado mediante métricas estándar, MAE (Mean Absolute Value) y RMSE (Error Cuadrático medio). Con estas métricas se puede llevar a cabo una comparativa entre ellos.

El modelo ARIMA fue entrenado con los parámetros óptimos que seleccionó automáticamente la función `auto_arima`, resultando en la configuración ARIMA(2,1,5), con un AIC: 419.140,027. Este modelo demostró ser competente en la predicción de la demanda energética univariante, aunque limitado al no incorporar variables exógenas.

El modelo SARIMA, con un coste computacional considerablemente mayor traducido en tiempos de espera, fue seleccionado analizando la serie, a través de un análisis de Autocorrelación y un test ADF (Dickey-Fuller). Finalmente, el modelo de menor AIC fue SARIMA (1,1,3)(1,1,1)[24].

AIC=379.163,291

MAE y RMSE: 3783,97 MW y 4396,85 MW

Aunque este modelo mejoró la representación de la estacionalidad, los errores continuarán siendo elevados frente a modelos más complejos.

Finalmente, el modelo XGBoost, que se fue haciendo más complejo por iteraciones incluyendo mejoras sucesivas, fue el más eficaz en términos de precisión y eficiencia computacional. Se incluyeron, a diferencia del resto de modelos, las variables exógenas de festividad, temperatura, día de la semana, hora del día, etc.

MAE y RMSE: 1626,85 MW y 2210,45 MW

Comparando los tres modelos en las tablas 4 y 5, se puede visualmente seleccionar el más adecuado para la serie temporal trabajada.

Modelo	AIC
<b>ARIMA(2,1,5)(0,0,0)[0]</b>	419.140,027
<b>SARIMA(1,1,3)(1,1,1)[24]</b>	379.163,291

*Tabla 4: Tabla comparativa ARIMA y SARIMA según AIC*

Entre los modelos ARIMA y SARIMA, el de menor AIC es el SARIMA, ya que este último incorpora de forma explícita la componente estacional de la serie, lo que resulta especialmente relevante en el caso de la variable dependiente que se trabaja; la demanda eléctrica. El modelo ARIMA identifica la tendencia general, y efectos a corto plazo a través de p y q, pero el hecho de no poder tener en cuenta el factor de la estacionalidad hace que no sea el modelo idóneo para predecir la demanda eléctrica.

En la tabla 5 se presentan las métricas de error absoluto medio (MAE) y raíz del error cuadrático medio (RMSE), que permiten comparar de manera directa el modelo tradicional SARIMA (mejor que ARIMA) frente al modelo avanzado de aprendizaje automático XGBoost.

Modelo	MAE	RMSE
<b>SARIMA</b>	3783,97 MW	4396,85 MW
<b>XGBoost</b>	1626,85 MW	2210,45 MW

*Tabla 5: Tabla comparativa SARIMA y XGBoost según errores*

Como se observa, XGBoost presenta mejores resultados tanto en MAE como en RMSE que SARIMA. Concretamente, con un MAE 57% menor y un RMSE de aproximadamente la mitad. Esto indica que el modelo XGBoost comete errores más pequeños (en promedio) al reflejar menor MAE y también limita mejor la magnitud de los errores extremos, por presentar menor RMSE.

El RMSE penaliza los errores grandes de forma cuadrática, por lo que tener un RMSE pequeño implica que XGBoost no solo es más preciso en términos medios, sino que también es menos propenso a cometer desviaciones graves en momentos puntuales, algo especialmente relevante cuando se trata de planificación energética.

El modelo XGBoost es mejor debido a varios factores, principalmente a la capacidad de incorporar variables exógenas como la festividad, ya que, como se ha estudiado al principio del proyecto, estos factores tienen gran influencia en la demanda eléctrica. En cambio, los modelos ARIMA y SARIMA son univariantes, por lo que no pueden considerarlas y se centran únicamente en los valores históricos de demanda para cada hora.

Por otro lado, el modelo SARIMA (y por lo tanto también ARIMA) es un modelo lineal, a diferencia de XGBoost, que es capaz de establecer relaciones no lineales entre las variables.

En conjunto, estos resultados refuerzan la idea de que los modelos basados en machine learning, bien ajustados y alimentados con información contextual, pueden ofrecer una mejora sustancial en la calidad de las predicciones respecto a los enfoques estadísticos tradicionales, especialmente en contextos complejos como el de la demanda energética.



## 8. Capítulo VIII: Explicabilidad y Análisis de resultados

En este contexto de uso creciente de modelos estadísticos y de machine learning para predicciones, la necesidad de que dichos modelos puedan ser comprensibles es cada vez más importante. Como se mencionó en el capítulo III, muchos de los modelos más precisos se emplean como cajas negras, especialmente aquellos basados en técnicas más avanzadas como XGBoost. Es decir, ofrecen predicciones, pero no es posible identificar qué factores han llevado a exactamente esos resultados.

Como se ha visto a lo largo del proyecto, la transparencia en las predicciones es una herramienta muy valiosa para anticipar resultados, a través de los análisis de sensibilidad, y para generar confianza en los modelos.

Por ende, en este capítulo, se busca emplear técnicas de interpretabilidad global y local que permitan describir la manera en la que cada uno de los modelos utiliza la información en sus predicciones. También se podrá comparar la interpretabilidad de modelos lineales como son ARIMA y SARIMA frente a XGBoost, que es un modelo no lineal. Finalmente, se validarán y comprobarán las explicaciones. Con ello, se pretende comprender y justificar las predicciones de cada uno de ellos sobre la demanda eléctrica.

### 8.1 Técnicas de explicabilidad: SHAP y LIME

#### 8.1.1 ARIMA

El modelo ARIMA es un modelo estadístico lineal, por lo que se pueden extraer conclusiones acerca de su explicabilidad a través de sus coeficientes. Cada término autorregresivo ( $p$ , AR) y de media móvil ( $q$ , MA) tiene un peso asociado y se puede hacer una interpretación sobre esas magnitudes.

En primer lugar, en la figura 39 se aprecia un patrón estacional bastante marcado, ya que se aprecian ciclos diarios o semanales en lugar de un trazo aleatorio. En la vista de la figura 39, debido a que se ha capturado un rango amplio de fechas, no se observa la amplitud de estos ciclos, pero sí es claro que los hay.

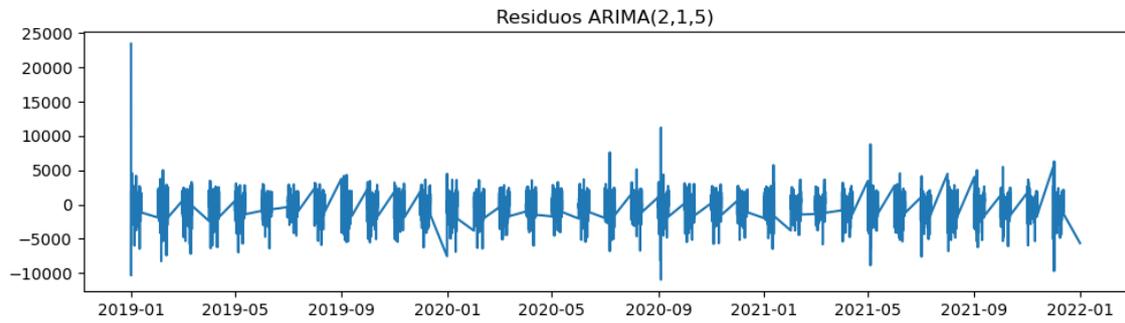


Figura 39: Gráfica de los residuos ARIMA(2,1,5)

En la figura 40, la función de autocorrelación ACF de los residuos, muestra dos ramas que resaltan respecto del resto, concretamente en los lags (o puntos temporales) 12 y 24, lo que estaría ilustrando un ciclo diario (aprox. 12 y 24h). Es decir, que existe autocorrelación entre los residuos.

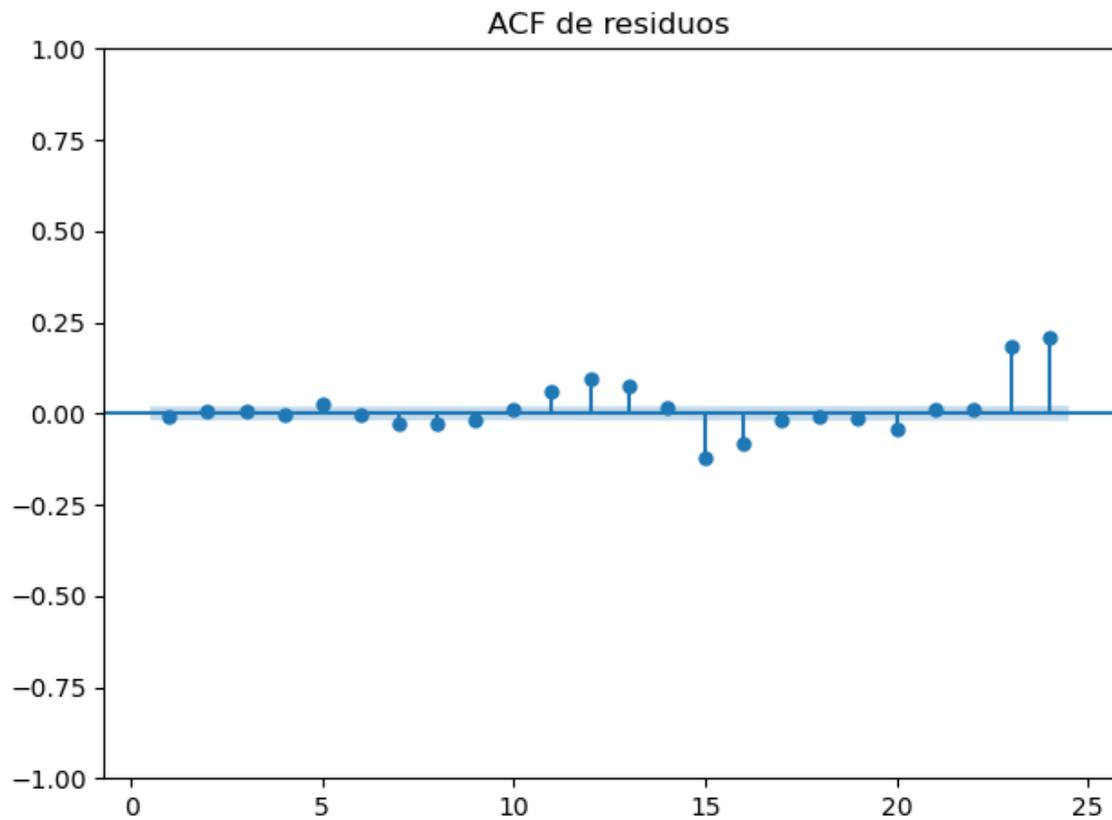


Figura 40: ACF de los residuos ARIMA

Por otro lado, se introduce SHAP en el modelo ARIMA para complementar su explicabilidad.

Para ello, se entrena un modelo wrapper: un modelo XGBoost (por ejemplo) que replica las predicciones del modelo ARIMA. Para ello, se requiere la instalación de la librería SHAP.

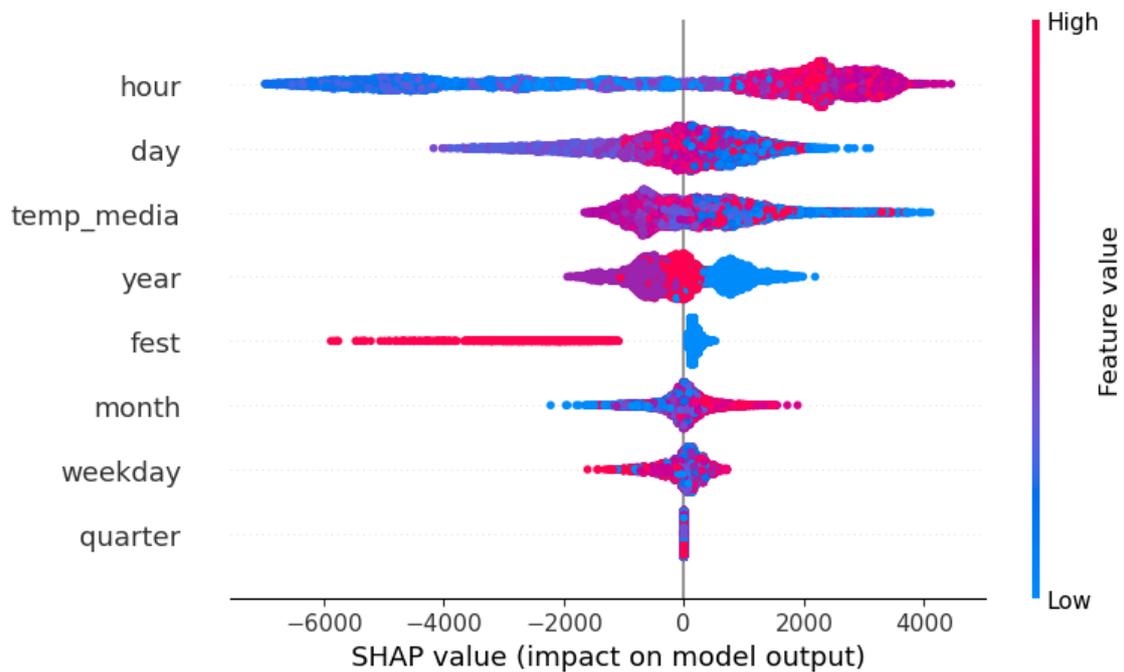


Figura 41: Resultados Summary Plot de SHAP

En la figura 39, se presenta un análisis gráfico tras aplicar SHAP al modelo que ordena las características de arriba abajo, según el orden de importancia. Por tanto, la variable más relevante y primera del ranking es la variable 'hora', seguida de la variable 'día'. Eso significa que, para este XGBoost que imita el comportamiento de ARIMA, la hora del día y el día del mes explican más variación en la demanda que la temperatura o el hecho de ser día festivo.

Por otro lado, este mismo gráfico muestra para cada punto, una posición y un color.

La posición horizontal, es decir, si está más a la derecha o hacia la izquierda, indica el impacto que tiene cada observación en la predicción. Es decir, los valores positivos, que

están más hacia la derecha, tienden a llevar la predicción al alza, en este caso: mayor demanda. Mientras que los que se encuentran más a la izquierda, la reducen. El color indica el valor original de la variable en esa observación, siendo rojo un valor alto (High, en el gráfico) y azul uno bajo (Low).

Analizando el caso de la variable 'hora', los puntos rojos son horas más altas (por la tarde y tarde noche, ya que el formato horario del dataset es 24h) y dichas horas aparecen en la derecha, demostrando que esas horas elevan la demanda de energía eléctrica. Por otro lado, los puntos azules (0:00, 1:00, y el resto de horas de la madrugada) reducen la demanda.

En la tabla de la figura 40, que muestra los primeros valores de SHAP para la muestra de entrenamiento, se observa numéricamente el comportamiento que se acaba de analizar.

	temp_media	fest	hour	day \
datetime2_local				
2019-01-01 01:00:00	544.215088	-2514.020508	-652.611633	251.732986
2019-01-01 02:00:00	447.369019	-2706.021729	-2702.438232	280.429565
2019-01-01 03:00:00	322.123596	-2655.254883	-3978.971680	111.376892
2019-01-01 04:00:00	336.613098	-2749.497559	-4874.825684	190.885895
2019-01-01 05:00:00	99.728828	-2827.988037	-5082.706543	221.850281
	month	weekday	quarter	year
datetime2_local				
2019-01-01 01:00:00	-987.946167	148.553055	0.0	725.624451
2019-01-01 02:00:00	-1068.565063	169.775146	0.0	667.673035
2019-01-01 03:00:00	-1249.484741	188.647949	0.0	596.268250
2019-01-01 04:00:00	-1256.144043	145.621033	0.0	604.052795
2019-01-01 05:00:00	-1356.369751	157.848846	0.0	634.549866

Figura 42: Primeras filas de valores SHAP

En este caso, se puede comprobar que un festivo siempre aporta un valor negativo y muy elevado en comparación al resto de valores de la tabla, lo que significa que reduce drásticamente la demanda, tal y como se ha estudiado en el capítulo II de este proyecto. La hora de madrugada también aparece con grandes valores SHAP negativos, mientras que la temperatura suma positivamente valores como 544, 447, 322... pero en una magnitud menor (posición 3 del ranking en la figura 39).

Lo interesante es observar cómo, a cada hora, cada una de las variables suma/resta una cantidad diferente a cada predicción. Son estos inputs los que enriquecen los análisis de modelos predictivos, y verdaderamente ofrecen una visión mucho más

### 8.1.2 SARIMA

Del mismo modo que se puede realizar una interpretación directa con el modelo ARIMA, también se puede con SARIMA. Sin embargo, para que el modelo presente una interpretabilidad semejante a aquella de modelos más complejos, también se puede aplicar LIME, entrenando un sustituto local que pueda aproximar el comportamiento de SARIMA alrededor de una predicción determinada. Para ello, se instala la librería lime.

Como se ha mencionado, LIME es un método local que se puede adaptar a series temporales. Con su aplicación a un momento concreto, se analiza qué rezagos empujan a la predicción del modelo al alza o a la baja. Es decir, se analiza en un momento concreto, qué valores en t-1, t-2 y t-24 hacen que el modelo SARIMA tienda a subir o bajar sus predicciones.

Para ello, dada la serie original se construye un dataset donde cada fila tiene tres columnas; el valor de la demanda en t-1, el valor de la demanda en t-2, en t-3, y el valor de la demanda en t-24 (un día antes a esa misma hora). Como las primeras 24 filas no tienen esos datos concretos, se eliminan. Se entrena el modelo SARIMA(1,1,3)(1,1,1)[24] a toda la serie, se selecciona un punto de interés y finalmente LIME crea variaciones sintéticas de esa fila y aprende un modelo lineal local que aproxima el comportamiento de SARIMA alrededor de esa instancia.

Los resultados se recogen a continuación en la figura 43

```
Valores de la instancia: {'lag_1': 25046.91666666667, 'lag_2': 26363.75, 'lag_24': 25638.83333333333}
```

```
Contribuciones LIME:
```

```
23960.37 < lag_1 <= 27633.92 : -2434.270
```

```
23960.37 < lag_2 <= 27634.08 : 242.651
```

```
23961.67 < lag_24 <= 27635.08 : -5.638
```

*Figura 43: Resultados al emplear el método LIME al modelo SARIMA*

En la primera fila de la figura 43, aparecen los valores anteriores al punto seleccionado. En ese momento, la demanda una hora antes fue de 25046,9166, dos horas antes fue de 26363,75 y un día antes valía 25638,8333.

El valor de la contribución del lag\_1 (la demanda una hora antes de la que se busca predecir) es de 2434,27 negativo, lo que quiere decir que tiene un impacto negativo muy fuerte, su valor hace que el modelo reduzca la predicción en 2434 unidades. Es la variable dominante en esta predicción.

El lag 2 (dos horas antes) aporta poco en positivo, 242,65. Y el lag 24, es decir, la demanda del día anterior a esa misma hora es el retardo que menos aporta y lo hace negativamente, siendo de -5,6, lo cual resulta prácticamente despreciable si lo comparamos con el resto de valores.

### 8.1.3 XGBoost

Este último, al ser un modelo basado en árboles de decisión, la técnica idónea es aplicar SHAP. De hecho, existe un método llamado TreeSHAP que permite obtener los valores de contribución de cada variable a cada predicción, sin ser computacionalmente ineficiente.

Por ende, se analiza a continuación este método para explicar cada predicción individual y también una interpretación global del modelo.

En primer lugar, en la figura 43 se observa la importancia global de las variables temperatura y festividad. La variable temperatura tiene una mayor influencia, según el gráfico, en las predicciones que la variable festividad. Las temperaturas altas suelen provocar un aumento significativo en la demanda eléctrica, mientras que los días festivos presentan un impacto más reducido y disperso.

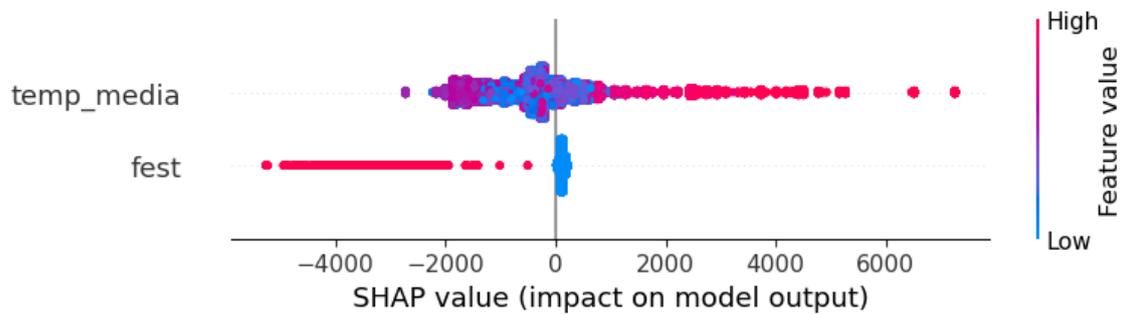


Figura 44: Gráfica 1 SHAP en modelo XGBoost

Por otro lado, en el gráfico de la figura 44 se puede apreciar el efecto no lineal de la temperatura. Concretamente, la variable temperatura presenta un efecto en forma de U, donde las temperaturas templadas se asocian a un consumo energético bajo y demandas elevadas ocurren en el momento de temperaturas muy bajas o muy altas, evidenciando el uso de calefacción o aire acondicionado.

El color indica el valor de la variable festividad (0, no festivo en azul y 1, festivo en rojo), lo cual no muestra un gran cambio de comportamiento entre festivos y no festivos para esta variable.

Este último apunte puede ser resultado de la simplificación que se hizo en un principio de solo tener en cuenta los días festivos de las comunidades con mayor número de habitantes. Esta aproximación puede estar provocando que la variable fest pierda impacto en relación a la demanda por ser poco precisa.

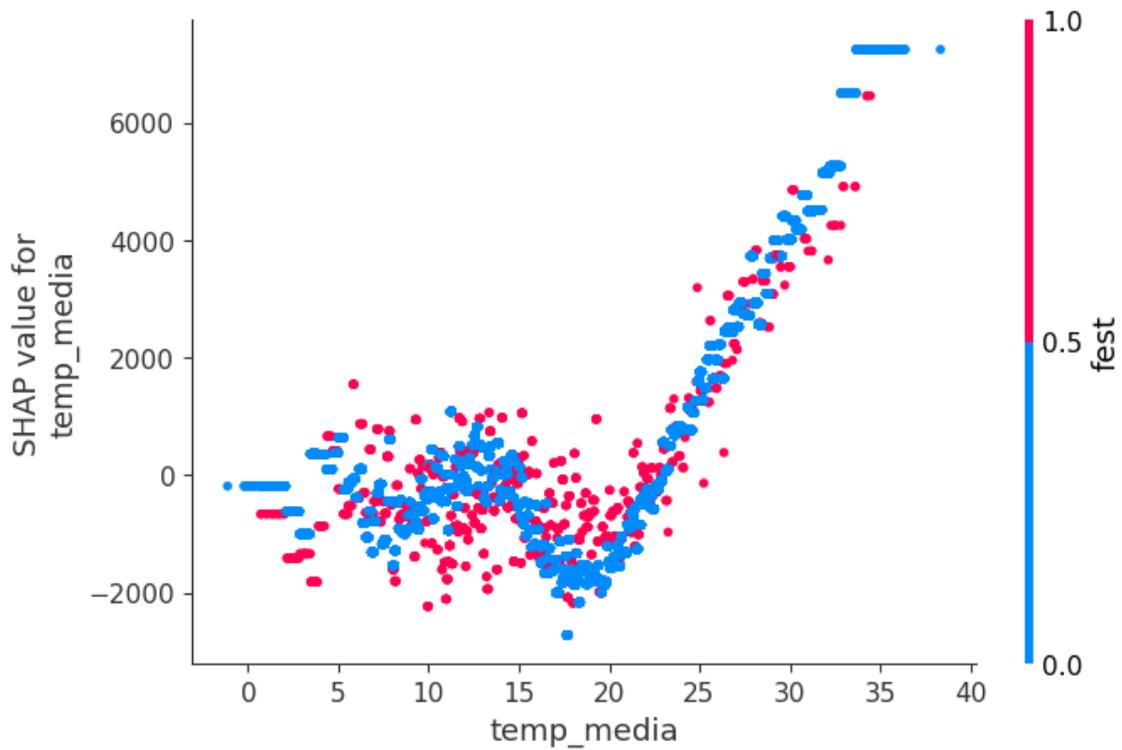


Figura 45: Gráfica 2 SHAP en modelo XGBoost

Finalmente, para comprobar la dependencia de valores pasados, se agregan tres variables nuevas de retardo, una la hora inmediatamente anterior, la otra 24 horas antes y la última 1 semana antes.

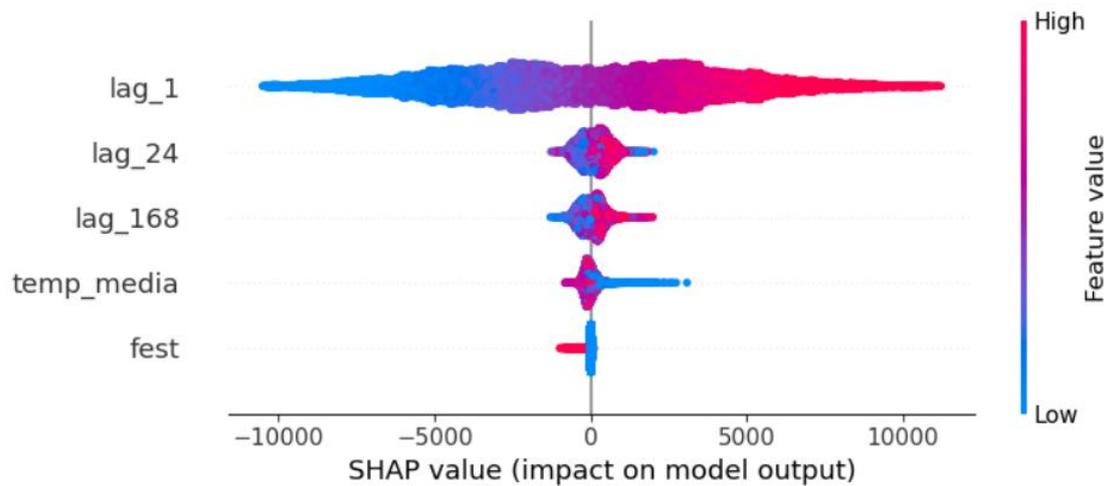


Figura 46: Gráfica 3 SHAP XGBoost con retardos

Como era previsible, la variable que más impacta, tanto positiva como negativamente, es el valor de la demanda una hora antes.

Por el contrario, temperatura y festividad, que en modelos anteriores sin retardos mostraban más peso relativo, ahora tienen un impacto menor. Esto indica que, al incorporar información temporal directa de la propia demanda pasada, el modelo logra captar mejor las dinámicas subyacentes, reduciendo la dependencia de variables exógenas.

## 8.2 Resultados y conclusiones

Este apartado recoge las principales conclusiones obtenidas tras la incorporación de métodos predictivos como SARIMA y XGBoost y técnicas de interpretabilidad avanzadas, centradas en el análisis mediante SHAP y LIME aplicado a modelos predictivos.

Desde un punto de vista metodológico, se ha comprobado que el modelo ARIMA no es capaz de capturar adecuadamente la estacionalidad diaria de la serie de demanda eléctrica. El análisis de residuos revela una clara persistencia a 12 y 24 periodos, así como la presencia de patrones cíclicos, lo cual vulnera el supuesto de 'ruido blanco' necesario para un modelo ARIMA correcto.

Para resolver esta limitación, se amplía a un modelo SARIMA, que incorpora un componente estacional explícito (por ejemplo, un término  $s=24$ ), y la integración de variables que modelicen el ciclo diario de forma directa. Esta integración hace que el modelo SARIMA presente resultados mejores ya que se asemeja más al comportamiento de la curva de demanda eléctrica.

Estas estrategias permitirían reducir la autocorrelación residual, mejorar el ajuste y, en consecuencia, la calidad de los pronósticos. En paralelo, se ha desarrollado un modelo XGBoost estructurado para simular el comportamiento de un ARIMA con capacidad multivariable. Sobre este modelo se ha aplicado el enfoque SHAP (SHapley Additive exPlanations) con el objetivo de interpretar sus decisiones. Este análisis ha validado varias de las hipótesis formuladas en un inicio:

Los patrones horarios y calendarios mensuales constituyen los principales impulsores de la demanda. Las variables temporales más generales (como mes o día de la semana) presentan mucha menor influencia.

Una de las aportaciones clave del trabajo ha sido precisamente la incorporación sistemática de métodos de explicabilidad al modelo predictivo. El análisis SHAP ha permitido no solo cuantificar la importancia de cada variable, sino también entender cómo contribuyen individualmente a las predicciones.

En una primera fase, el modelo se entrenó únicamente con variables exógenas como la temperatura media y el indicador de festivo. El gráfico de importancia global mostró que la temperatura era el principal factor explicativo, y su gráfico de dependencia reflejaba una relación no lineal en forma de U: tanto las temperaturas bajas como las elevadas incrementan la demanda, mientras que los entornos térmicos moderados la reducen, como se planteó al principio del estudio.

En segundo lugar, al aplicar el método LIME al modelo SARIMA se evidenció el papel tan relevante que juega en las predicciones el valor de la demanda una hora antes, del mismo modo que se pudo observar la poca relevancia que tenía la variable de la demanda 24 horas antes del momento de la predicción, en términos relativos (comparado con la influencia del lag\_1).

En tercer lugar, el último modelo que se entrenó. Se enriqueció el conjunto de datos con variables de retardo, que corresponden a la demanda registrada una hora, un día y una semana antes. Esta ampliación mejoró significativamente el rendimiento del modelo. El nuevo análisis SHAP confirmó que lag\_1 es la variable con mayor impacto predictivo, reflejando la inercia natural del consumo eléctrico. Las variables lag\_24 y lag\_168 también resultaron relevantes, lo que demuestra la presencia de patrones diarios y semanales en la serie.

En consecuencia, las variables exógenas pasaron a un segundo plano, aunque mantuvieron cierto valor explicativo en condiciones extremas (por ejemplo, temperaturas muy altas o festivos especiales), pero se evidenció el valor de conocer los valores de demanda inmediatamente anteriores al momento de predicción para hacerlo con mayor precisión.

## 9. Capítulo IX: Estudio de viabilidad económica e impacto del proyecto

Este último capítulo se divide en dos secciones complementarias. Por un lado, se aborda la viabilidad económica del proyecto, evaluando los recursos invertidos y su eficiencia desde una perspectiva académica y tecnológica. Por otra, se examina el impacto que el trabajo genera en el contexto medioambiental e industrial, destacando su alineación con los Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030.

### 9.1 Viabilidad económica

En esta sección se detalla el análisis económico del proyecto, siguiendo la metodología aplicada a los proyectos de ingeniería.

El objetivo del trabajo ha sido el desarrollo de los modelos ARIMA, SARIMA y XGBoost para la predicción de la demanda eléctrica y un análisis de explicabilidad posterior con los métodos SHAP y LIME.

Para analizar la viabilidad económica de dicho proyecto, conviene desglosar cada categoría del presupuesto. Puesto que se trata de un proyecto de corta duración y que no requiere de material ni entornos específicos, la inversión está principalmente vinculada al tiempo dedicado en el desarrollo del proyecto y algunos recursos digitales.

Así, el desglose resulta de la manera recogida en la tabla 6.

Categoría	Explicación	Coste en €
Tiempo	Alumna: 200h x 25€/h	5.000
	Director: 20h x 50€/h	1.000
Materiales	Ordenador	200
	Amortización de las 200h	

	Licencias software y servicios online	200
IVA estimado (21% aprox.)	21% de 6.400	1.344
TOTAL		7.744€

Tabla 6: Inversión en euros

Como puede observarse, el componente principal es el coste del trabajo personal, que en un contexto formativo se interpreta como una inversión en capital humano. Las herramientas utilizadas son de código abierto o de bajo coste, y el trabajo se ha desarrollado en un entorno virtual (Anaconda, JupyterLab), lo que ha permitido optimizar al máximo los recursos. También se ha hecho uso del paquete de Microsoft Office, que en este caso no supone un gasto por tener acceso a ello desde el usuario de la universidad sin coste adicional.

Aunque este proyecto no genera ingresos directos, puede evaluarse su rentabilidad desde una perspectiva de coste por unidad. En este caso, se considera que se han desarrollado y analizado 5 modelos predictivos, ARIMA, SARIMA, XGBoost y las técnicas SHAP y LIME. El coste normalizado (coste total entre unidades de desarrollo, en este caso modelos) sería de aproximadamente 1.548,8 €/modelo, cifra coherente con la naturaleza del proyecto y su impacto formativo y técnico.

La planificación ha sido la presentada en la figura , dedicando unas 200 horas aproximadamente al proyecto de una manera no lineal: al final se ha dedicado más tiempo por día que al principio.

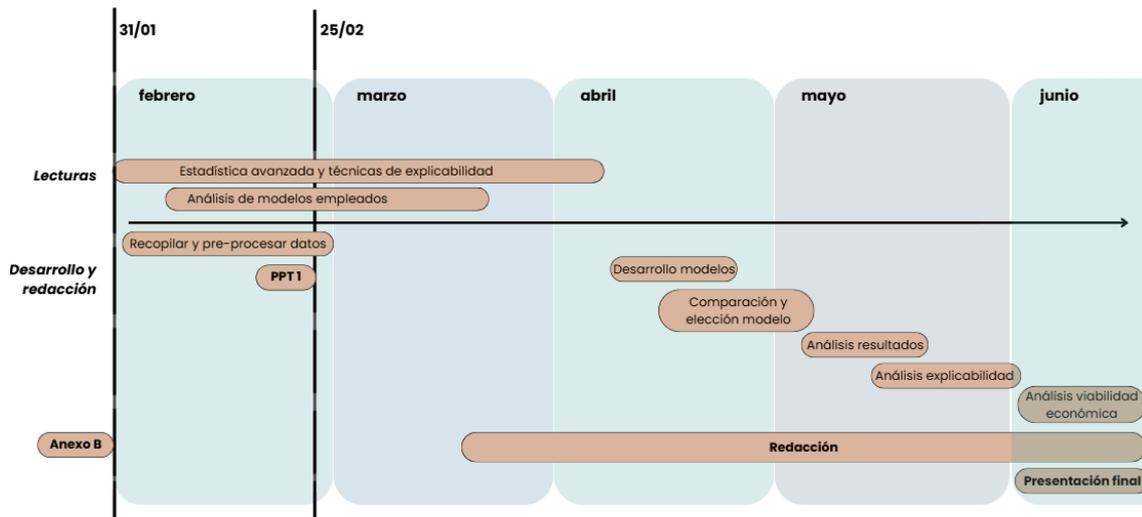


Figura 47: Planificación del proyecto. (Elaboración propia)

## 9.2 Impacto del proyecto y alineación ODS

Finalmente se analiza el alcance e implicaciones del trabajo en términos de sostenibilidad, tecnología y medioambiente. Se mencionan y se profundiza en los Objetivos de Desarrollo Sostenible con los que se alinea de manera directa: ODS 7 Energía asequible y no contaminante, ODS 9 Industria, Innovación e infraestructura y ODS 13 Acción por el clima.

### ODS 7: Energía asequible y no contaminante

Este proyecto se alinea con el Objetivo 7 ya que contribuye a una generación de energía más planificada, con mejor calidad y de manera más eficiente gracias a un mayor entendimiento sobre las predicciones de la demanda eléctrica a través de la aplicación de técnicas de explicabilidad.

La anticipación precisa del consumo eléctrico facilita una generación programada que permita un ajuste preciso entre demanda y generación. Esto permite una mayor colaboración de energías renovables en la generación total de energía, contribuyendo directamente a una transición energética limpia y sostenible.



Figura 48: ODS 7. (ONU, <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>)

#### ODS 9: Industria, innovación e infraestructura

El uso de técnicas avanzadas como XGBoost y de técnicas de explicabilidad para comprender el fundamento de las predicciones representa una innovación en el sector energético. Este proyecto fomenta la inclusión de tecnologías novedosas y su conocimiento a través de métodos de explicabilidad en los sistemas de operación de redes eléctricas, lo que refuerza las infraestructuras digitales del país y de las empresas del sector.

El uso de herramientas de explicabilidad como SHAP y LIME mejora la transparencia y confianza en los modelos, ampliando su uso en otros sectores más allá de la salud o las finanzas y haciendo posible su adopción en entornos críticos como el mercado eléctrico o la planificación de la generación.



Figura 49: ODS 9. (ONU, <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>)

#### ODS 13: Acción por el clima

Al fomentar una energía asequible y no contaminante, y promover soluciones tecnológicas sostenibles que fomentan la innovación en la industria energética, se impacta de manera directa y positivamente sobre el clima. Una predicción precisa de la demanda permite planificar la generación de fuentes de energía renovables y por ende reducir el uso de generación ineficiente de respaldo, que a menudo se basa en combustibles fósiles. De esta forma, el proyecto ayuda a minimizar las emisiones de gases de efecto invernadero, contribuyendo a mitigar el cambio climático.



Figura 50: ODS 13. (ONU, <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>)



proyecto

Capítulo IX: Estudio de viabilidad económica e impacto del

## Referencias

- [1] Energies, "A review of electricity demand forecasting techniques: from classical models to artificial intelligence," *Energies*, vol. 17, no. 2, p. 367, 2024. [Online]. <https://doi.org/10.3390/en17020367>
- [2] Red Eléctrica de España, "Evolución de la demanda eléctrica," [Online]. <https://www.sistemaelectrico-ree.es/es/informe-del-sistema-electrico/demanda/evolucion-demanda>
- [3] Red Eléctrica de España, "Demanda eléctrica y actividad económica," [Online]. [https://www.ree.es/sites/default/files/downloadable/demanda-electrica-actividad-economica\\_0.pdf](https://www.ree.es/sites/default/files/downloadable/demanda-electrica-actividad-economica_0.pdf)
- [4] Ecooo, "El PIB y la demanda eléctrica dejan de tener una relación directa," [Online]. <https://ecooo.es/blog/el-pib-y-la-demanda-electrica-dejan-de-tener-una-relacion-directa/>
- [5] Red Eléctrica de España, "Atlas INDEL," [Online]. [https://www.ree.es/sites/default/files/downloadable/atlas\\_indel\\_ree.pdf](https://www.ree.es/sites/default/files/downloadable/atlas_indel_ree.pdf)
- [6] Iberdrola, "Olas de calor y energía limpia: en un futuro con picos de consumo," [Online]. <https://www.iberdrola.es/blog/autoconsumo/olas-de-calor-y-energia-limpia-en-un-futuro-con-picos-de-consumo>
- [7] T. Gómez, "Modelos ARIMA, SARIMA y método de Box-Jenkins," Medium, [Online]. <https://datasciencepe.substack.com/p/modelos-arima-sarima-y-metodo-de>
- [8] AI Blog, "Shap vs. LIME: Herramientas de Interpretabilidad de Modelos Comparadas," [Online]. [https://iartificial.blog/de/aprendizaje/shap-vs-lime-herramientas-de-interpretabilidad-de-modelos-comparadas/#goog\\_rewarded](https://iartificial.blog/de/aprendizaje/shap-vs-lime-herramientas-de-interpretabilidad-de-modelos-comparadas/#goog_rewarded)
- [9] "LIME vs SHAP: A Comparative Analysis of Interpretability Tools," [Online]. <https://www.markovml.com/blog/lime-vs-shap>

- [10] PMC, "Practical guide to SHAP analysis," [Online].  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10072891/>
- [11] Wikipedia, "Valor de Shapley," [Online].  
[https://es.wikipedia.org/wiki/Valor\\_de\\_Shapley](https://es.wikipedia.org/wiki/Valor_de_Shapley)
- [12] T. Gómez, "SHAP: SHapley Additive exPlanations," Medium, [Online].  
<https://tutegomeze.medium.com/shap-shapley-additive-explanations-9a0d0e7e63d5>
- [13] Minitab, "Partial Autocorrelation Function (PACF) – Interpret the results," [Online].  
<https://support.minitab.com/es-mx/minitab/help-and-how-to/statistical-modeling/time-series/how-to/partial-autocorrelation/interpret-the-results/partial-autocorrelation-function-pacf/>
- [14] Instituto para la Diversificación y Ahorro de la Energía (IDAE), *Tendencias y políticas de eficiencia energética en España*. Edición 2024. Ministerio para la Transición Ecológica y el Reto Demográfico, 2024. [Online].  
<https://www.odyssee-mure.eu/publications/national-reports/espana-eficiencia-energetica.pdf>
- [15] M. Filonchyk and A. Dolzhenko, "Temperature and Residential Electricity Demand for Heating and Cooling," *Energies*. 2022. [Online].  
<https://www.mdpi.com/2225-1154/10/10/142>
- [16] OMIE, "Precios del mercado diario - Marzo 2017," Operador del Mercado Ibérico de Energía, 2017. [Online]. <https://www.omie.es>
- [17] OMIE, "Precios del mercado diario - Marzo 2024," Operador del Mercado Ibérico de Energía, 2024. [Online]. <https://www.omie.es>
- [18] Red Eléctrica de España (REE), *Informe del sistema eléctrico español 2017*, Madrid, 2017. [Online]. <https://www.ree.es/es/publicaciones>

- [19] Red Eléctrica de España (REE), *Avance del informe del sistema eléctrico español 2024*, Madrid, 2024. [Online]. <https://www.ree.es/es/publicaciones>
- [20] Red Eléctrica de España, "Portal de datos abiertos," [Online]. Available: <https://www.ree.es/es/datos>
- [21] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, Wiley, 2015.
- [22] J. Masís, *Interpretable Machine Learning with Python*, 2022.
- [23] X. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 785–794, 2016. [Online]. <https://dl.acm.org/doi/pdf/10.1145/2939672.2939785>
- [24] R. Harode, "XGBoost: A Deep Dive into Boosting," *Medium*, 2020. [Online]. <https://medium.com/sfu-csmp/xgboost-a-deep-dive-into-boosting-f06c9c41349>
- [25] Agencia Estatal de Meteorología, "Datos climáticos históricos," AEMET OpenData. [Online]. <https://opendata.aemet.es>
- [26] ENTSO-E, "Transparency Platform," [Online]. <https://transparency.entsoe.eu>
- [27] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774, 2017. [Online]. [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf)
- [28] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge*

- Discovery and Data Mining*, pp. 1135–1144, 2016. [Online].  
<https://doi.org/10.1145/2939672.2939778>
- [29] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56–67, 2020. [Online]. <https://doi.org/10.1038/s42256-019-0138-9>
- [30] ONU, "Objetivos de Desarrollo Sostenible," [Online].  
<https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>
- [31] Fundación Endesa, "Historia de la electricidad," [Online].  
<https://www.fundacionendesa.org/es/educacion/endesa-educa/recursos/historia-de-laelectricidad>
- [32] C. Molnar, *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*, 2nd ed., Leanpub, 2022. [Online].  
<https://christophm.github.io/interpretable-ml-book/>
- [33] S. M. Lundberg and S.-I. Lee, "Explainable machine-learning predictions for the prevention of hypoxaemia during surgery," *Nature Biomedical Engineering*, vol. 2, pp. 749–760, 2018. [Online].  
<https://doi.org/10.1038/s41551-018-0304-0>
- [34] D. Slack et al., "Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods," *Proc. AAAI/ACM Conf. AI, Ethics, and Society*, pp. 180–186, 2020. [Online]. <https://doi.org/10.1145/3375627.3375830>
- [35] L. S. Shapley, "A value for n-person games," in *Contributions to the Theory of Games*, vol. 2, H. W. Kuhn and A. W. Tucker, Eds., Princeton University Press, 1953, pp. 307–317.

- [36] H. P. Young, "Monotonic solutions of cooperative games," *International Journal of Game Theory*, vol. 14, no. 2, pp. 65–72, 1985. [Online]. <https://doi.org/10.1007/BF01769275>
- [37] R. Mulla, "Time Series Forecasting," YouTube, 2023. [Online]. [https://www.youtube.com/watch?v=vV12dGe\\_Fho&t=372s](https://www.youtube.com/watch?v=vV12dGe_Fho&t=372s)
- [38] T. Hong, P. Pinson, and S. Fan, "Global Energy Forecasting Competition 2012," *International Journal of Forecasting*, vol. 30, no. 2, pp. 357–363, 2014. [Online]. Available: <https://doi.org/10.1016/j.ijforecast.2013.07.001>
- [39] OpenWeather, "Weather data for business," [Online]. <https://openweathermap.org/>
- [40] UCI Machine Learning Repository, "Individual Household Electric Power Consumption Data Set," [Online]. <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption>
- [41] R. Weron, "Electricity price forecasting: A review of the state-of-the-art with a look into the future," *International Journal of Forecasting*, vol. 30, no. 4, pp. 1030–1081, 2014. [Online]. <https://doi.org/10.1016/j.ijforecast.2014.08.008>
- [42] F. Ziel and B. Liu, "Econometric and machine learning approaches to electricity load forecasting: A review," *Wiley Interdisciplinary Reviews: Energy and Environment*, vol. 10, no. 4, e406, 2021. [Online]. <https://doi.org/10.1002/wene.406>