Proyecto Fin de Grado
Grado en Ingeniería de las Tecnologías Industriales, Mención en Electricidad

Comparativa de Métodos de Aprendizaje Automático Aplicados a la Predicción del Precio del Mercado Eléctrico Diario

Autor: Eugenio Fraguas Valero

Tutor: Jesús Manuel Riquelme Santos

Dpto. Ingeniería Eléctrica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021

Proyecto Fin de Grado
Grado en Ingeniería de las Tecnologías Industriales

Comparativa de Métodos de Aprendizaje Automático Aplicados a la Predicción del Precio del Mercado Eléctrico Diario

Autor:
Eugenio Fraguas Valero

Tutor:
Jesús Manuel Riquelme Santos
Profesor titular

Dpto. de Ingeniería Eléctrica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2021

Proyecto Fin de Grado: Comparativa de Métodos de Aprendizaje Automático Aplicados a la Predicción del Precio del Mercado Eléctrico Diario

Autor: Eugenio Fraguas Valero

Tutor: Jesús Manuel Riquelme Santos

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2021

El Secretario del Tribunal

A mi familia

A mis amigos

A mis maestros

Agradecimientos

Al terminar los estudios, uno suele recordar el camino que le llevó hasta el final. Quiénes estuvieron en el camino, quiénes ayudaron ya sea con apuntes, con explicaciones, o con apoyo moral o logístico. En mi caso, la lista es enorme.

En pimer lugar, he de agradecer a mis padres el apoyo incondicional y la paciencia que he disfrutado. No es común que los padres aguanten tantos años sin quejarse de lo que uno está tardando en terminar los estudios, y esa fé finalmente se ha visto recompensada.

A mis hermanos les agradezco el respeto que han tenido hacia mí y hacia mis lagas jornadas de estudios, sobre todo durante el confinamiento, cuidándose de no molestarme durante todo el tiempo que permanecía en mi habitación.

Al resto de mi familia he de agradecer la confianza que han tenido siempre en mí, pues muchas veces han creído más que mí mismo en que sería capaz de lograrlo, y me lo demostraron con cariño siempre.

Agradezco enormemente también a mis profesores, exigentes y profesionales, la capacidad que han tenido de transmitir el conocimiento y el interés por la profesión, en especial a mi tutor. De no ser por algunos de ellos, puedo asegurales que no estaría escribiendo estas palabras hoy.

Mis compañeros de clase han supuesto también un pilar fundamental donde apoyarme también, compartiendo apuntes y conocimientos, por lo que he de agradecerles sin duda estos años de compañerismo. Espero encontrarme a todos ellos en un futuro disfrutando de una próspera carrera pofesional.

Por últmo, no podría terminar sin recordar a mis amigos, a todos, sin los cuales esto no habría sido posible. Algunos compañeros de estudios, otros simplemente amigos, pero en cualquier caso responsables de risas, de consuelo, de relativizar lo malo y acrecentar lo bueno. Sé que este logro lo harán suyo, al igual que míos son sus éxitos también.

Eugenio Fraguas Valero

Sevilla, 2021

Resumen

Tras la reforma del sistema eléctrico del año 1997 que liberalizó el mercado en España, el precio de la electicidad pasó a estar determinado por una subasta pública. De esta forma, la energía en nuestro país pasó a ser comeciada mediante un modelo marginalista en el cual el precio viene determinado por el corte entre las curvas de oferta y demanda.

Ante la cada vez mayor complejidad del mercado debido al aumento de los consumidores, las nuevas tecnologías de generación renovable, y las nuevas formas de consumo derivadas de la movilidad eléctrica y la generación distribuída, se hace necesario disponer de herramientas que ofrezcan certidumbre sobe la evolución del mercado en el futuro, de forma que organismos reguladores, instituciones gubernamentales e inversores privados, entre otros, disfruten de mayor seguridad a la hora de tomar futuras decisiones en lo referente al mercado eléctrico de nuestro país.

En los últimos años, la ciencia de datos, sustentada en una mayor capacidad de almacenamiento y análisis de información, ha experimentado un crecimiento exponencial llegando a estar presente en numerosos y diferentes sectores. Gran parte de los métodos de aprendizaje automático resultan de especial utilidad en la predicción de valores futuros basándose en datos históricos, convirtiéndolos por tanto en magníficos recursos para la optimización y el análisis de nuestro mercado eléctrico.

En el presente trabajo de fin de grado se detalla un estudio comparativo entre diferentes métodos de aprendizaje automático con el objetivo de determinar cuál de ellos resulta más preciso a la hora de predecir el precio diario futuro del megavatio hora.

Para ello, en primer lugar se recopilaron los siguientes datos horarios históricos comprendidos en el periodo que abarca del 01/01/2021 a las 1:00 h al 03/05/2021 a las 0:00 h: *Precio (€/MWh), Demanda (MWh), Generación de Carbón (MWh), Generación de Ciclo Combinado (MWh), Generación de Cogeneración (MWh), Generación Eólica (MWh), Generación Hidráulica (MWh), Generación Nuclear (MWh), Generación Solar Fortovoltaica (MWh), Generación Solar Térmica (MWh), Intercambios Energéticos con las Islas Baleares (MWh) e Intercambios Energéticos Internacionales (MWh).*

Una vez procesados los datos para su posterior análisis, se les realizó un estudio multivariante mediante varios métodos con el objetivo de seleccionar aquellas combinaciones de atibutos que, reduciendo el número de variables a procesar, maximizasen el resultado obtenido. Los diferentes métodos de selección de variables empleados fueron los siguientes: *Correlación de Pearson, Información Mutua, Selección Regresiva, y LASSO.*

Posteriormente, se probaron las diferentes combinaciones de atributos con los métodos de aprendizaje automático listados a continuación: *Regresión Lineal, Árbloes de decisión, K-Vecinos Próximos, Máquinas de Soporte Vectorial y Redes Neuronales.*

Tras comparar todos los métodos con las diferentes combinaciones, se llegó a la conclusión de que el mejor método de predicción en todos los casos resultó ser el de los *Árboles de Decisión,* siendo la combinación de atributos que mejor ajustaba la predicción la obtenida mediante la *Correlación de Pearson.*

Abstract

After 1997's electricity system reform that liberalized the market in Spain, the price of electricity was determined by a public auction. In this way, energy in our country began to be traded through a marginalist model in which the price is determined by the cut between the supply and demand curves.

Given the rising complexity of the market due to the increase in consumers, new renewable generation technologies, and new forms of consumption derived from electric mobility and distributed generation, it is necessary to have tools that offer certainty about the evolution of the market in the future, so that regulatory bodies, government institutions and private investors, among others, enjoy greater security when making future decisions regarding the electricity market of our country.

In recent years, data science, supported by a greater capacity for storing and analyzing information, has experienced exponential growth, becoming present in many different sectors. Many of the machine learning methods are especially useful in predicting future values based on historical data, thus making them magnificent resources for the optimization and analysis of our electricity market.

In this thesis, a comparative study between different machine learning methods is detailed with the aim of determining which of them is more accurate when predicting the future daily price of megawatt-hour.

To do this, first of all, the following historical hourly data were collected for the period from 01/01/2021 at 1:00 a.m. to 05/03/2021 at 0:00 a.m.: *Price (€ / MWh)* , *Demand (MWh), Coal Generation (MWh)*, *Combined Cycle Generation (MWh)*, *Cogeneration Generation (MWh)*, *Wind Generation (MWh)*, *Hydraulic Generation (MWh)*, *Nuclear Generation (MWh)*, *Solar Generation (MWh)*, *Solar Thermal Generation (MWh)*, *Energy Exchanges with the Balearic Islands (MWh)* and *International Energy Exchanges (MWh)*.

Once the data had been processed for subsequent analysis, a multivariate study was performed using various methods, with the aim of selecting those combinations of attributes that, by reducing the number of variables to be processed, would maximize the result obtained. The different methods of selection of variables used were the following: *Pearson Correlation*, *Mutual Information*, *Backward Selection*, and *LASSO*.

Subsequently, the different combinations of attributes were tested with the machine learning methods listed below: *Linear Regression*, *Decision Trees*, *K-Neighboring Neighbors*, *Vector Support Machines* and *Neural Networks*.

After comparing all the methods with the different combinations, it was concluded that the best prediction method in all cases turned out to be *Decision Trees*, being the combination of attributes that best adjusted the prediction the one obtained through the *Pearson Correlation* method.

Índice

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

1 INTRODUCCIÓN

1.1 Motivación

Big Data, *Machine Learning* o *Redes Neuronales* son términos que rápidamente han pasado a formar parte de nuestro vocabulario diario. De la noche a la mañana la ciencia de datos ha ocupado un lugar importante en la mayoría de las actividades económicas y sociales de nuestro entorno, impulsada por un enorme desarrollo de las tecnologías, el cual ha permitido por primera vez en la historia de la humanidad recopilar y procesar cantidades ingentes de información.

El análisis de datos es empleado en la actualidad en tareas tan dispares como la predicción del compotamiento del mercado bursátil, el análisis de tácticas en diferentes deportes como el fútbol, o en asuntos de salud pública como por ejemplo en el estudio de la evolución de la pandemia de Covid-19.

Los diferentes métodos de recopilación y análisis de datos actuales se centran en la optimización y en la predicción. Así, en los casos mencionados, las tácticas deportivas serán analizadas con el objetivo de optimizar diferentes aspectos del juego, como por ejemplo, el ratio de tiros a puerta frente a goles anotados en el fútbol. Sin embargo, en lo referente al análisis de la evolución de la pandemia provocada por el virus SARS-CoV-2, las diferentes técnicas de aprendizaje automático tendrán como objetivo principal el de anticipar una evolución de la enfermedad en la población mundial.

El sector eléctrico resulta extremadamente complejo de operar y analizar. Involucra variables macroeconómicas como el precio de materias primas o cotizaciones como las de los mercados diario e intradiario, además de infinidad de requisitos técnicos que garanticen correcto funcionamiento del sistema. Cuenta también con una gran cantidad de datos recopilables y analizables que abarcan desde el estado y topología de la red, hasta los precios registrados en el pool, pasando por las lecturas de los nuevos contadores inteligentes.

Todo ello convierten al sistema eléctrico en un sujeto perfecto para la implementación de diferentes métodos de análisis de datos, debido a la gran cantidad de sistemas y procesos que se verían indudablemente beneficiados de las ventajas que estas técnicas ofrecen.

1.2 Alcance

La motivación fundamental del proyecto es la de poner el foco, dentro de los diferentes campos en los que el análisis de datos es implementado en el sistema eléctrico, en la predicción de precios del mercado eléctrico diario.

Entre los pasos necesarios para realizar dicho estudio se encuentran, además de los referentes al propio análisis, la recopilación de los datos empleados para el estudio y el aprendizaje en el uso de algunas de las herramientas empleadas para llevar a cabo las diferentes técnicas de *Machine Learning* como el lenguaje de programación *Python*, así como la comprensión de los diferentes aspectos técnicos y teóricos que involucran a los diferentes métodos de selección y análisis.

En este Trabajo de Fin de Grado se ha realizado pues, un estudio comparativo de diferentes técnicas de aprendizaje automático con el objetivo de determiner cuál es la que ofrece una predicción del precio diario del mercado eléctrio más precisa. Para ello, se ha realizado una selección de atibutos previa para reducir el volumen de datos a manipular, pasando a ser elegidas mediante un análisis multivariente aquellas combinaciones de variables más significativas para el objeto del estudio, en nuestro caso cuatro. Tras determinar las diferentes combinaciones de atributos, los distintos métodos de aprendizaje automático elegidos, a saber, *Regresión Lineal, Árboles de Decisión, K-Vecinos Próximos, Máquinas de Soporte Vectorial y Redes Neuronales*, fueron probados para las cuatro combinaiones diferentes.

Para evaluar los resultados, se emplearon diferentes estimadores como el *Error Cuadrático Medio (MSE)* o el

Error Medio Absoluto (MAE).

Tras realizar el análisis completo, se determinó que el método que mejor desempeñó la tarea fue el de los *Árboles de Decision*, siendo la combinación más precisa la determinada por el método de selección de atributos mediante la *Correlación de Pearson.*

2 SISTEMA ELÉCTRICO ESPAÑOL

El desarrollo progresivo del hombre depende vitalmente de la invención; es el producto más importante de su cerebro creativo.

- Nikola Tesla -

Para llegar a realizar un análisis correcto mediante la ciencia de datos, es necesario además de conocer en profundidad las diferentes técnicas y algoritmos que involucran dicha rama, estudiar el sector en el que se va a desarrollar dicho informe. Así, una parte fundamental del proceso de estudio y tratamiento de datos no es otro que su visualización y comprensión. De nada sirve obtener resultados si éstos no son comprendidos a posteriori por los técnicos oportunos. Por ello, antes de abordar en profundidad lo referente al aprendizaje automático, es necesario comprender en primer lugar el sector en el que se ha desarrollado el estudio. En este proyecto, pues, se dedica un primer apartado destinado a definir y explicar la estructura del sistema eléctrico español y el funcionamiento de su mercado.

2.1 Estructura del Sistema Eléctrico Español

Desde 1996, en el marco de la Unión Europea se vienen adoptando medidas dirigidas a la liberalización de los diferentes sistemas eléctricos europeos con el fin de establecer un sistema más competitivo y flexible, con precios de suministro basados en el mercado. Dicho proceso se fue materializando de forma progresiva a través de diferentes directivas de liberalización.

La primera directiva fue adoptada en 1996 con el objetivo de que los mercados eléctricos de los estados miembro fuesen liberalizados antes de 1998 (Ciucci, 2020).

El Segundo paquete energético fue adoptado en el año 2003, y permitía que los consumidores eligiesen libremente a su proveedor de gas y electricidad (Ciucci, 2020).

En 2009 se aprobó un tercer bloque que introducía modificaciones en el segundo con el objetivo de liberalizar los mercados interiores de gas y electricidad (Ciucci, 2020).

Por ultimo, en el año 2019 se aprobó un cuarto paquete energético acompañado además de tres reglamentos sobre la electricidad, sobre la protección de riesgos, y para la creación de la Agencia Europea para la Cooperación de los Reguladores de la Energía (ACER) respectivamente. Éste incluía nuevas normativas para el mercado eléctrico con el objetivo de favorecer las tecnologías renovables y la inversión privada en el sector, estableciendo además un límite para que las centrales eléctricas puedan recibir subvenciones como mecanismos de capacidad (Ciucci, 2020).

Como resultado de las progresivas normativas comunitarias, en la actualidad el mercado eléctrico español es un mercado liberalizado y dividido en cuatro partes diferentes: Generación, Transporte, Distribución y Comercialización. Esta división implica que no puedan existir empresas presentes en más de un sector del sistema. En el caso de las antiguas grandes empresas monopolistas, dicha normativa supuso su división en varias sociedades, cada una dedicada exclusivamente a uno de los sectores y sin posibilidad alguna de cooperación entre ellas. A modo de ejemplo nos encontramos el caso de la *Empresa Nacional De Electricidad Sociedad Anónima* (ENDESA), la cual en la actualidad se encuentra constituida por más de cien sociedades diferentes entre las que destacan, por ejemplo, *Enel Green Power* y *Endesa generación Nuclear* (generación), *Endesa Red* (distribución), y *Endesa Soluciones* y *Endesa Energía* (comercialización) (Endesa, 2019).

El sector de la generación se encuentra compuesto por empresas productoras de electricidad. Así, la generación es el nivel donde se encuentran las diferentes centrales eléctricas, presas hidráulicas y parques fotovoltaicos y eólicos, todos ellos en manos de diferentes empresas cuya actividad económica principal consiste en su explotación.

El transporte queda en manos de *Red Eléctrica de España (REE),* empresa pública encargada de la explotación de la red eléctrica de transporte (subestaciones, líneas y demás activos de la red), cuyos niveles de tensión se encuentran acotados entre los cuatrocientos kilovoltios y los doscientos veinte kilovoltios.

La distribución eléctrica corre a cargo de diferentes compañías propietarias de la red de distribución eléctrica a lo largo del territorio. Los niveles de tensión de distribución oscilan entre los ciento treinta y dos kilovoltios y los sesenta y seis kilovoltios para alta tensión, de sesenta y seis kolovoltios a un kilovoltio en media tensión, y de un kilovoltio a cuatrocientos voltios en baja tensión Dependiendo del lugar geográfico, la propiedad de la red de distribución corresponderá a una compañía diferente.

La comercialización es el área del sistema donde diferentes empresas ofrecen contratos de suministro a sus clientes, que es libre de elegir la compañía que desee. Dichas sociedades pagan tarifas por el acceso a las redes de distribución necesarias para suministrar la energía que ofertan. La energía a comercializar es adquirida en la subasta pública, proceso que se detallará en el siguiente apartado.

2.2 El Mercado Eléctrico Español

2.2.1 La Factura Eléctrica

El precio final de la energía que ha de abonar el consumidor es la suma de varias tasas y costes destinados a sufragar las distintas actividades, procesos y componentes que conforman el sistema eléctrico español. El importe final de la factura, pues, será la suma de una serie de componentes que se detallan a continuación:

- **Tarifa de acceso.** Consiste en una aglomeración de costes regulados que incluyen el coste de las redes, la compensación ante la disparidad del coste de producción en las regiones insulares, anualidades para recuperar el deficit de tarifas, y el fomento de las energías renovables (CNMC, 2021).

- **Impuestos.** Incluyen el impuesto sobre el valor absoluto (IVA) y el impuesto sobre la electricidad. (CNMC, 2021).

- **Alquiler del equipo de medida.** Importe destinado a sufragar el coste del contador. (CNMC, 2021).

- **Márgen de beneficio de la comercializadora.** (CNMC, 2021).

- **Coste de la energía.** Se encuentra compuesto por los pagos por capacidad, los pagos a los operadores del sistema y del mercado (REE y OMIE), el coste de los servicios de ajuste y el coste de la energía en el mercado diario. Éste ultimo será el objeto de studio de este proyecto (CNMC, 2021).

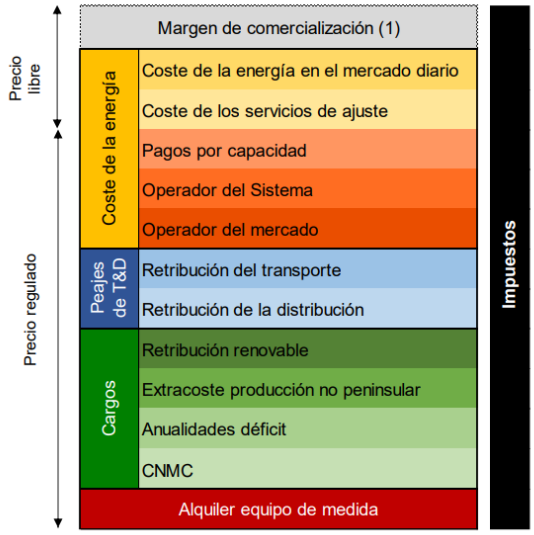

Figura 2.1. Componentes de la factura de la luz. Fuente: (CNMC, 2021).

2.2.2 El Mercado Eléctrico

2.2.2.1 El Pool Eléctrico

El mercado mayorista de la electricidad en nuestro país recibe el nombre de Pool eléctrico. En él generadores y comercializadores realizan sus correspondientes ofertas, de forma que los primeros ofertan por el mínimo que están dispuesto a cobrar por generar electricidad, y los segundos por el máximo que están dispuestos a pagar por ella. El precio final será el determinado por el corte entre las curvas de precios de generadores y comercializadores. Así, los productores que estuviesen dispuestos a generar por debajo del precio de corte serán aquellos que entren en el mercado, obteniendo mayor beneficio del previsto inicialmente. Análogamente, aquellos comercializadores que pujasen un precio mayor al finalmente establecido entrarán en el mercado obteniendo la energía a un precio menor al que inicialmente propusieron.

El precio del mercado diario será, pues, el precio marcado por el corte de las curvas mencionadas enteriormente. Existen también mercados secundarios destinados a cubrir posibles contingencias del sistema que escapan al alcance de este proyecto.

2.2.2.2 OMIE

- OMIE es el operador del mercado eléctrico designado (NEMO según terminología europea) para la gestion del mercado elétrico diario e intradiario de la Peninsula Ibérica. Es el encargado de recibir las ofetas de compraventa de energía en la subasta diaria, además de regular todo el proceso. (OMIE, 2021)

.

2.2.2.3 Mercado Diario

Según OMIE:

> "El mercado diario también llamado acoplamiento único diario (SDAC, por sus siglas en ingles), como parte integrante del mercado de producción de energía eléctrica, tiene por objeto llevar a cabo las transacciones de energía eléctrica mediante la presentación de ofertas de venta y adquisición de energía

eléctrica por parte de los agentes del mercado para las venticuatro horas del día siguiente. Este mercado, acoplado con Europa desde el año 2014, es una de las piezas cruciales para conseguir el objetivo del Mercado Interior de la Energía Europeo.

Todos los días del año a las 12:00 CET, se lleva a cabo la sesión del mercado diario en la que se fijan los precios y energías de la electricidad en toda Europa para las veinticuatro horas del día siguiente. El precio y el volumen de energía en una hora determinada se establecen por el cruce entre la oferta y la demanda, siguiendo el modelo acordado y aprobado por todos los mercados europeos que actualmente es de aplicación en España, Portugal, Alemania, Austria, Bélgica, Bulgaria, Croacia, Eslovaquia, Eslovenia, Estonia, Francia, Holanda, Hungría, Irlanda, Italia, Letonia, Lituania, Luxemburgo, Finlandia, Suecia, Dinamarca, Noruega, Polonia, Reino Unido, República Checa y Rumania.

Los agentes compradores y vendedores que se encuentren en España o en Portugal presentarán sus ofertas al mercado diario a través de OMIE, que es el único NEMO designado en dichos países. Sus ofertas de compra y venta son aceptadas atendiendo a su orden de mérito económico y en función de la capacidad de interconexión disponible entre las zonas de precio. Si en una cierta hora del día la capacidad de la interconexión entre dos zonas es suficiente para permitir el flujo de electricidad resultante de la negociación, el precio de la electricidad en esa hora será el mismo en ambas zonas. Si, por el contrario, en esa hora la interconexión se ocupa totalmente, en ese momento el algoritmo para la fijación del precio da como resultado un precio diferente en cada zona. Este mecanismo descrito para la formación del precio de la electricidad se denomina acoplamiento de mercados.

Los resultados del mercado diario, a partir de la libre contratación entre agentes compradores y vendedores representan la solución más eficiente desde el punto de vista económico, pero dadas las características de la electricidad, se necesita que sea también viable desde el punto de vista físico. Por ello, una vez obtenidos estos resultados se remiten al Operador del Sistema para su validación desde el punto de vista de la viabilidad técnica. Este proceso se denomina gestión de las restricciones técnicas del sistema y asegura que los resultados del mercado sean técnicamente factibles en la red de transporte. Por tanto, los resultados del mercado diario pueden sufrir pequeñas variaciones como consecuencia del análisis de restricciones técnicas que realiza el Operador del Sistema, dando lugar a un programa diario viable".

(OMIE, 2021).

2.2.2.4 Cálculo del Precio

En el año 2009 varios mercados europeos pusieron en marcha la iniciativa "*Price Coupling of Regions" (PCR),* la cual desarrolló un algoritmo de cálculo de precios llamado *Euphemia.* Dicho software persigue la maximización de las ganancias tanto de compradores como de vendedores, garantizando a la vez la optimización del uso de la capacidad disponible en las interconexiones.

Para dicho cometido, *Euphemia* considera las curvas agregadas en escalón. Así, como se mencionó anteriormente, las empresas generadoras hacen sus ofertas (cantidad de energía y precio), y las comercializadoras y demás clientes demandan la energía que necesitan a un precio determinado.

Dichas ofertas son entonces ordenadas de forma creciente en el caso de los generadores, y decreciente en el caso de los compradores. El punto de corte de ambas curvas se denomina punto de casación, punto que maximiza las ganancias para ambas partes. El punto de casación establece por tanto el precio de la energía para esa hora determinada. De esta forma, toda energía ofertada y demandada por debajo del precio de casación será adjudicada a ese precio, dejando fuera al resto de ofertas que superen ese importe. (Miralles, 2017).

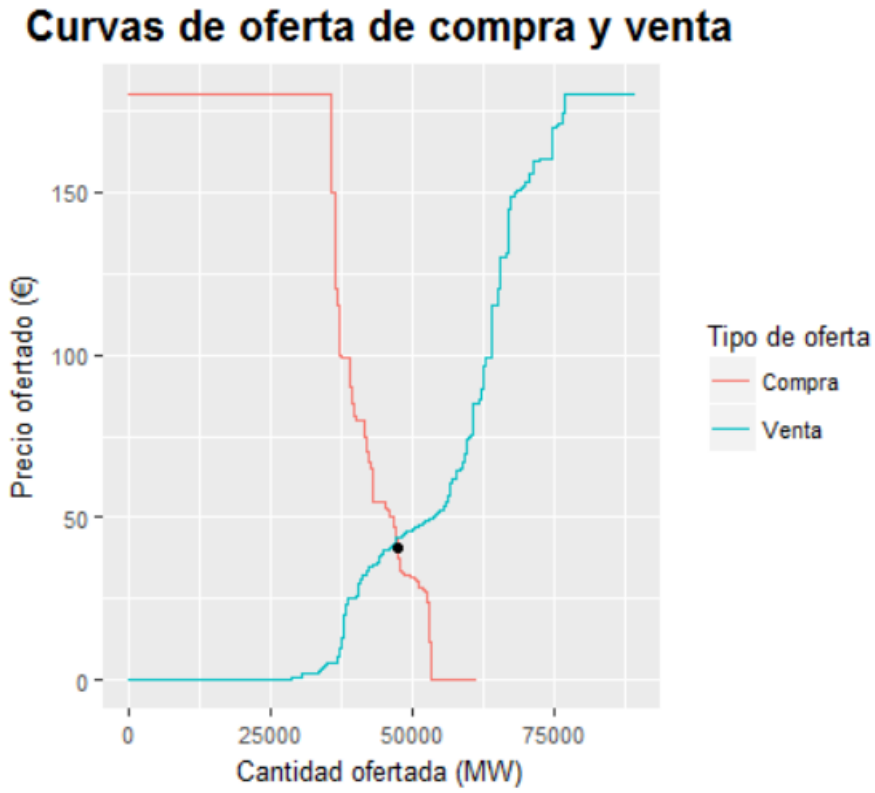

Figura 2.2 Curvas agregadas de compra/venta y precio de la energía para el 22/08/2016 a las 0:00. Fuente: (Miralles, 2017).

3 ESTADO DEL ARTE

Una computadora puede ser llamada "inteligente" si logra engañar a una persona haciéndole creer que es un humano.

- Alan M. Turing -

Tras haber descrito la jerarquía del sistema eléctrico español, así como la composición y el funcionamiento de su mercado, en este capítulo se abordan los aspectos técnicos tanto del lenguaje de programación empleado para el análisis objeto de este proyecto, *Python*, como de las diferentes técnicas de aprendizaje automático probadas.

Se comienza por una introducción a los conceptos y términos básicos de la materia, continuando con una presentación y descripción de *Python,* y con un desglose de las variables más relevantes para elestudio. Por último, se concluye con una exposición de las diferentes técnicas de análisis empleadas tanto para la selección de atributos como para la predicción.

3.1 Conceptos Básicos de Ciencia de Datos y Aprendizaje Automático

La ciencia de datos ha experimentado en la última década un crecimiento exponencial. Como resultado, en la actualidad convivimos con tecnologías que resultan familiares que, sin embargo, hace una década habrían resultado para muchos arte de magia. Dicho boom tecnológico ha generado una terminología completamente nueva que, sin ser bien comprendida por el públio general, está sin embargo en boca de todos. El objeto de este proyecto es el de analizar y comparar diferente técnicas de aprendizaje automático con el objeivo de determinar cuál ofrece más exactitud a la hora de predecir el precio del mercado diario. Por ello, antes de abordar la materia es necesario aclarar las diferente confusiones que puedan surgir a la hora de diferenciar términos.

3.1.1 Introducción Histórica

Aunque pudiera parecer que la ciencia de datos cuenta con una historia breve a sus espaldas, lo cierto es que, al contrario que lo que dicta la creencia popular, esta rama de la ciencia lleva existiendo varias décadas.

El primer concepto que aparece en la órbita de la ciencia de datos es el de *Business Intelligence*, el cual hace referencia a la relación existente entre la recopilación de datos para su posterior análisis y toma de decisiones correspondiente. Fue por primera vez empleado por Hans Peter Luhn, ingeniero de *IMB* en 1958, aunque no sería popularizado hasta finales de la década de los ochenta. Con el tiempo, la *Business Intelligence* quedó resumida a un análisis descriptivo de los datos que no involucra la búsqueda de patrones y tendencias que deriven en un análisis descriptivo (Niño, 2015).

Es en esa analítica predictiva donde surgen los términos *Data Mining* y *Data Science.*

Fue John W. Tukey quien en 1962 acuñó por primera vez el término en su artículo “El futuro sobre el análisis de datos”. En él, Tukey lo definía como “aquellos procedimientos para analizar datos, técnicas para interpretar resultados, formas de planificar la recopilación de datos para hacer más sencillo el proceso y, toda la maquinaria y los resultados de las estadísticas matemáticas que se aplican al análisis de datos.”. (B12, 2019).

En 1977 la Asociación Internacional de Computación Estadística afirmó que “La misión de este organismo es vincular la metodología estadística tradicional, la tecnología informática moderna y el conocimiento de expertos en el dominio para convertir los datos en información y conocimiento”, dando un paso más hacia la vinculación de las técnicas tradicionales de análisis con la potencia computacional que comenzaban a ofrecer los primeros

ordenadores. (B12, 2019).

Sin embargo, no sería hasta 2001 cuando William S. Cleveland propondría en su artículo *"Ciencia de Datos: un plan de acción para expandir las áreas técnicas del campo de la estadística"* que la ciencia de datos pasase a ser una disciplina independiente de la estadística. (B12, 2019).

Pese a las numerosas publicaciones realizadas durante décadas, en la mayoría de los casos la ciencia de datos se limitó al plano teórico. No fue hasta la segunda década de los años 2000 cuando comenzaron a desarrollarse de manera práctica diferentes técnicas de análisis, alimentadas ya por los poderes de almacenamiento y de cómputo necesarios para procesar tales cantidades de información. La llegada de la nube ofreció a los científicos de datos la posibilidad de almacenar y acceder de forma remota a cantidades ingentes de datos, ampliando las capacidades de la propia ciencia. Además, la posibilidad de fragmentar el problema para ser analizado por varios ordenadores en paralelo supuso una revolución en la capacidad de análisis, incrementando exponencialmente la complejidadad de los problemas que podían ser resueltos.

No solo la capacidad de almacenamiento y análisis fueron determinantes para el desarrollo de esta disciplina. La recolección de la información es otro pilar fundamental. La irrupción masiva de los dispositivos inteligentes (teléfonos, tablets, televisiones…) permitió por vez primera recopilar datos sobre multiples disciplinas diferentes (preferencias, geolocalización, consumo, estado de salud…) de forma continua.

La ciencia de datos es pues, un concepto que cuenta con varias décadas a sus espaldas, pero cuya noción por parte del público general data de hace poco más de un lustro, debido a que no fue hasta entonces cuando los tres pilares que la sustentan, teoría estadística, poder de almacenamiento y tratamiento de datos y recolección de información, fueron desarrollados ampliamente.

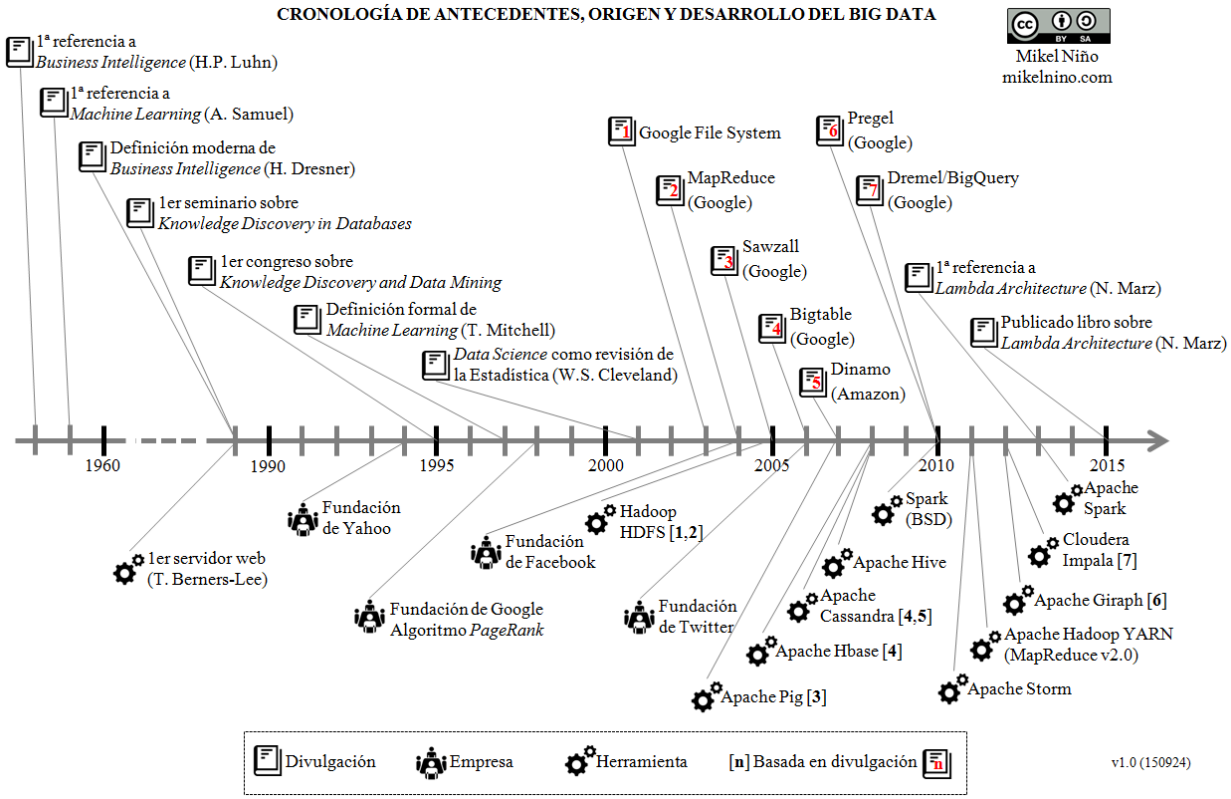

Figura 3.1 Cronología de antecedentes y desarrollo del *Big Data.* Fuente: (Niño, 2015)

3.1.2 Data Science

Data Science es la traducción literal del castellano "ciencia de datos". Muchas veces el término es confundido con *Big Data* o con *Data Mining*, términos que como veremos a continuación se encuentran relacionados, pero que en ningún casos son sinónomos. La ciencia de datos es la disciplina que aborda el estudio y procesamiento de los datos. En palabras del *MBIT School*:

> "Data Science es el término utilizado para el proceso que busca extraer grandes cantidades de datos para

determinar patrones repetitivos. Esto ayuda a organizar y controlar todos los aspectos variables de una organización, como los costos, la competencia y el mercado.

En sí, se encarga de estudiar el origen de la información, lo que representa y las formas que existen para emplearla a beneficio de cualquier proyecto".

(MBIT School, 2020)

3.1.3 Data Mining

El *Data Mining* o minería de datos es una subcategoría dentro de la ciencia de datos. Su objetivo es el de extraer patrones y comportamientos de grandes cantidades de datos con el objetivo de extraer conclusiones. (IEBS, 2019).

3.1.4 Big Data

El *Big Data* es el encargado del almacenamiento de grandes cantidades de información. Al referirnos a él, lo hacemos a las grandes cantidades de datos existentes de algún sector en concreto, los cuales podrán ser utilizados por la ciencia de datos para elaborar un análisis basado en ellos. (MBIT School, 2020).

Las diferencias entre el *Data Science* y el *Big Data* son notables (MBIT School, 2020):

- El *Data Science* es una herramienta pensada para extraer información del *BigData*.
- El *Big Data* cuenta con una capacidad de almacenamiento de datos que, si bien el *Data Science* hace uso de ellas, no las posee.

3.1.5 Machine Learning

El *Machine Learning* o aprendizaje automático es una disciplina diferente a las mencionadas anteriormente. Se encarga del desarrollo de sistemas que aprenden automáticamente identificando patrones y, que son capaces de predecir comportamientos futuros. Además su independencia implica su mejoría y optimización sin necesidad de intervención humana alguna. (González, s.f.).

Los métodos y técnicas empleadas para el desarrollo de este proyecto pertenecen al ámbito del *Machine Learning.*

Los algoritmos de *Machine Learning* suelen clasificarse en función de su tipo de aprendizaje:

- **Algoritmos de aprendizaje supervisado** son aquellos a los que se les proporciona una serie de datos conocidos con entradas y salidas, de forma que aprenden a obtener la relación entre ambos. Son los algoritmos predominantes en tareas relacionadas con la predicción de valores contínuos como los precios. Un ejemplo de algoritmo con aprendizaje supervisado son los algoritmos regresivos como el de regresión lineal, empleado en el studio realizado en este proyecto. (APD, 2019).
- **Algoritmos de aprendizaje sin spuervisar** son aquellos en los que los datos son proporcionados sin resultado o intervención humana alguna. Al no disponer de solución como en el caso del aprendizaje supervisado, el algoritmo desarrollará las relaciones mediante la información disponible. Así, el algoritmo interpretará grandes cantidades de datos para describir su estuctura. Esto suele implicar la organización de los datos en grupos. Un ejemplo de aprendizaje no supervisado son los algoritmos de clasificación, como por ejemplo los de recomendación de plataformas de contenido audiovisual en *streaming* o tiendas online. (APD, 2019)
- **Algoritmo de aprendizaje por refuerzo** según la redacción de APD, consiste en:

  "El aprendizaje por refuerzo se centra en los procesos de aprendizajes reglamentados, en los que se proporcionan algoritmos de aprendizaje automáticos con un conjunto de acciones, parámetros y valores finales.

  Al definir las reglas, el algoritmo de aprendizaje automático intenta explorar diferentes opciones y

posibilidades, monitorizando y evaluando cada resultado para determinar cuál es el óptimo.

En consecuencia, este sistema enseña la máquina a través del proceso de ensayo y error. Aprende de experiencias pasadas y comienza a adaptar su enfoque en respuesta a la situación para lograr el mejor resultado possible".

Tras haber definido los diferentes conceptos que componen la disciplina, se muestra un diagrama de Venn que refleja su jerarquía y la relación:

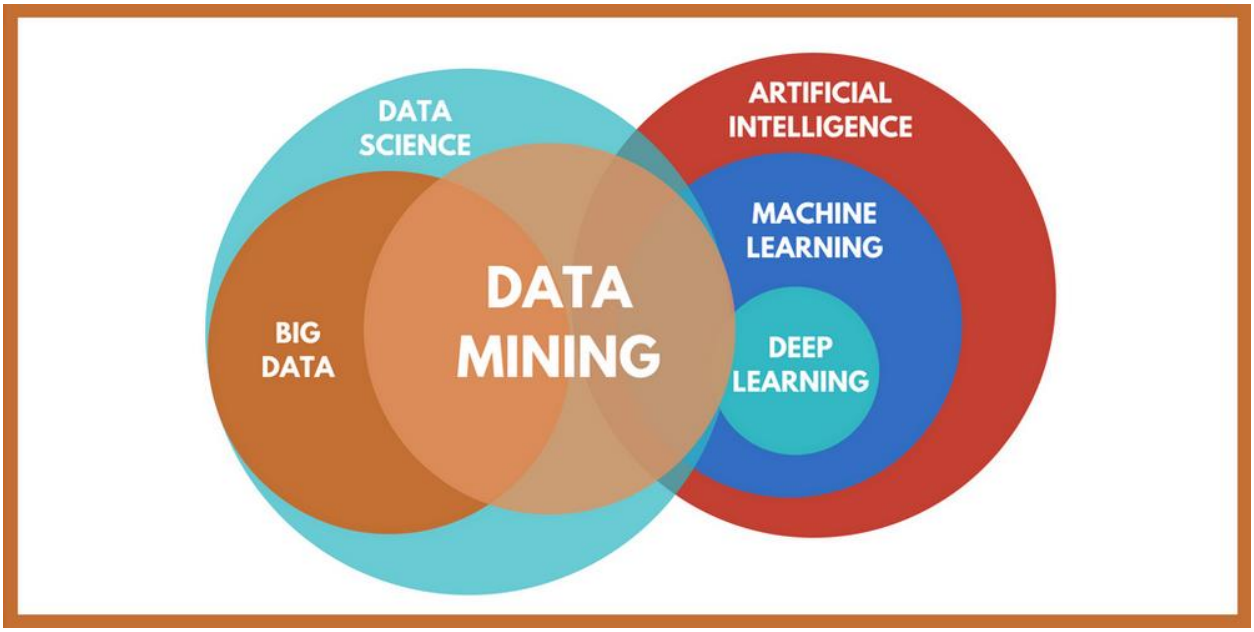

Figura 3.2 Relación entre diferentes campos relacionados con la ciencia de datos. Fuente: (DataEvo, 2018)

3.1.6 Python

"Python es un lenguaje de programación interpretado de tipado dinámico cuya filosofía hace hincapié en una sintaxis que favorezca un Código legible. Se trata de un lenguaje de programación multiparadigma disponible en varias plataformas". (Programo Ergo Sum, s.f.).

Que sea un **lenguaje interpretado** significa que no necesita ser compilado para ejecutar el programa, detectándose los errores en tiempo de ejecución. (Programo Ergo Sum, s.f.)

Es **multiparadigma**, lo que le hace soportar programación functional, imperativa, y orientada a objetos.

Su **tipado dinámico** implica que las variables son comprobadas también en tiempo de ejecución.

Se encuentra disponible para *Windows*, *Linux* y *MAC*, y es completamente gratuíto.

3.1.6.1 Librerías Significativas para el Análisis de Datos

Python es un lenguaje de programación que goza de una gran versatilidad, el cual es empleado para el desarrollo de aplicaciones móviles, programación de microcontroladores, y un largo etcétera. Es, junto a *R*, el lenguaje de programación más empleado en el entorno de *Data Science* y *Machine Learning*.

El hecho que convierte a *Python* en un lenguaje tan polivalente es el uso de librerías, paquetes con métodos, funciones y objetos preprogramados con el objetivo de agilizar y facilitar tareas y dotar de herramientas extra al software. Las librerías de *Python* empleadas para el desarrollo de este proyecto se detallan a continuación:

- ***Pandas*:** es una librería que dota al software con capacidad de cargar diferentes archivos de datos (.xlm, .csv, .json…) ubicados en diferentes escritorios, e incluso webs, en un tipo de variable denominado *Data Frame.* Un *Data Frame* podría entenderse como una matriz de m x n cuyas columnas son series

de datos, normalmente encabezados por un título (p.e. una serie de precios), y cuyas filas son normalmente una única variable (p.e. tiempo). Los Data Frames llevan asociados además diferentes métodos que les permiten ser manipulados, añadiendo o eliminando filas o columnas, cambiando los tipos de variables, sustituyendo ciertos valores por otros, etc. Esto les convierte en un elemento realmente práctico a la hora de trabajar con grandes cantidades de datos.

Out[6]:

	Fecha	Precio	Demanda	Carbón	Ciclo Combinado	Cogeneración y Resto	Eólica	Hidráulica	Nuclear	Solar FV	Solar Térmica	Intercambios Baleares	Intercambios Internacionales
0	2021-01-01T00:00:00+01:00	54.56	24722.833	260.667	2519.167	2400.333	6883.667	6225.667	7118.500	8.667	0.000	-109.000	-1116.167
1	2021-01-01T01:00:00+01:00	52.80	23571.500	261.000	2523.500	2366.167	6978.500	5776.333	7116.833	8.167	0.000	-100.000	-1905.500
2	2021-01-01T02:00:00+01:00	49.95	21491.000	258.167	2133.167	2357.667	7229.333	4715.000	7116.500	8.167	0.000	-100.000	-2771.667
3	2021-01-01T03:00:00+01:00	45.22	19709.167	258.500	1775.500	2353.333	7878.000	3081.500	7117.500	7.667	0.000	-100.000	-3228.333
4	2021-01-01T04:00:00+01:00	42.91	18791.667	258.000	1730.833	2355.500	8718.667	2093.000	7118.167	8.000	0.000	-100.000	-3935.667
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2946	2021-05-03T19:00:00+02:00	87.07	27627.333	423.833	3810.833	3517.000	3862.000	3710.500	6953.667	2207.833	725.000	-176.500	2005.333
2947	2021-05-03T20:00:00+02:00	91.33	28845.500	423.667	4187.000	3532.833	3720.000	4853.500	6986.333	711.833	581.167	-227.500	3456.833
2948	2021-05-03T21:00:00+02:00	91.69	30452.167	423.833	4517.000	3536.333	3294.000	6365.833	7019.833	26.500	365.333	-232.667	4547.333
2949	2021-05-03T22:00:00+02:00	88.01	28735.000	424.167	4399.667	3532.833	2901.667	5550.500	7044.667	6.333	299.667	-183.000	4162.333
2950	2021-05-03T23:00:00+02:00	86.33	26216.500	415.833	4142.667	3508.667	2888.333	3954.833	7046.667	5.667	296.167	-109.167	3481.167

2951 rows × 13 columns

Figura 3.3 Ejemplo de *Data Frame*. Fuente: elaboración propia

- ***Numpy*:** en ciertos lenguajes de programación como *Matlab* o *C++* existen ciertos tipos de variable denominados *arrays*. Éstos no son más que vactores que contienen varios elementos del mismo tipo, pudiendo ser de una o más dimensiones. *Python* no incorpora de forma natural la posibilidad de trabajar con *arrays*, de modo que debe ser añadida de forma externa a través de la librería *Numpy.*
- ***Plotly* y *Matplotlib***: contienen las funciones y métodos necesarios para la representación gráfica de funciones tanto en dos como en tres dimensiones. Permiten la creación de todo tipo de gráficos, desde rectas de regresión hasta histogramas.

Out[18]: <AxesSubplot:>

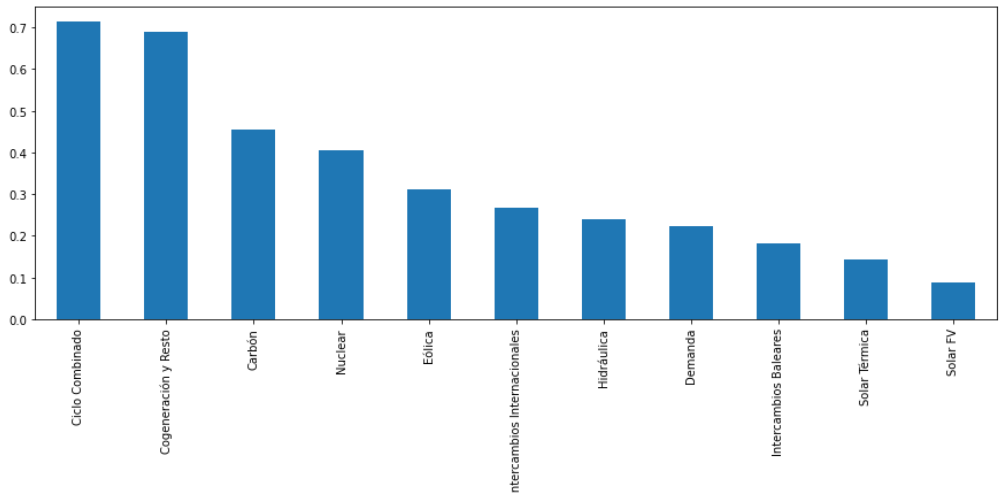

Figura 3.4 Ejemplo de histograma realizado con *Matplotlib*

- ***Scikit-learn:*** incluye métodos y algoritmos de aprendizaje automático.
- ***Keras y Tensorflow*:** librerías destinada al deep learning y a la constucción de redes neuronales.

3.2 Técnicas de Análisis

Tras haber definido y listado las principales librerías de Python empleadas en el análisis de datos, a continuación se desarrollarán los diferentes métodos empleados para la selección de variables y la predicción.

3.2.1 Selección de Atributos

La gran mayoría de algoritmos de *Machine Learning* requieren que se les suministren diferentes variables de las que depende el atributo a predecir. Sin embargo, al trabajar con numerosas variables, algunas pueden resultar redundantes, otras irrelevantes, y en cualquier caso, una cantidad elevada de datos suministrados dificultará el desempeño del método empleado, ralentizándolo y sobreentrenándolo en algunos casos.

3.2.1.1 Sobreentrenamiento

El sobreentrenamiento de un modelo, *overerfitting* en inglés, es fruto de un exceso de datos suministrados al mismo. Éste fenómeno implica un peor desempeño del modelo a la hora de su implementación, pues habrá aproximado tanto a los datos empleados para su entrenamiento, que su precision se verá mermada al proporcionarle datos reales para elaborar una predicción.

Pongamos por ejemplo el caso de un modelo de regresión lineal. En este caso el modelo se elaborará empleando unos datos de entrenamiento, a partir de los cuales se obtendrá la recta que major aproxime a dichos puntos. Si el modelo se sobreentrenase, la recta de regresión aproximaría de manera realmente precisa a los datos que han servido para la construcción del modelo, pero al suministrarle nuevos datos, la curva obtenida no aproximaría al nuevo set de datos, por lo que la predicción no sería del todo precisa.

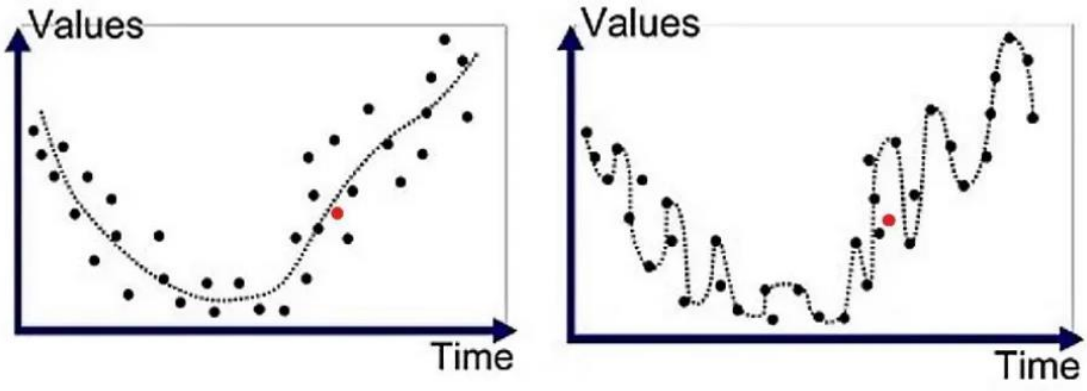

Figura 3.5 Modelo bien entrenado frente a modelo sobreentrenado. Fuente: (7 Hidden Layers, s.f.)

3.2.1.2 Infraentrenamiento

Por contraposición el infraentrenamiento, *underfitting,* se producirá si un modelo no es entrenado con información suficiente, es probable que desarrolle una aproximación pobre y poco precisa debido a que existen varias variables relvantes que no han sido tenidas en cuenta.

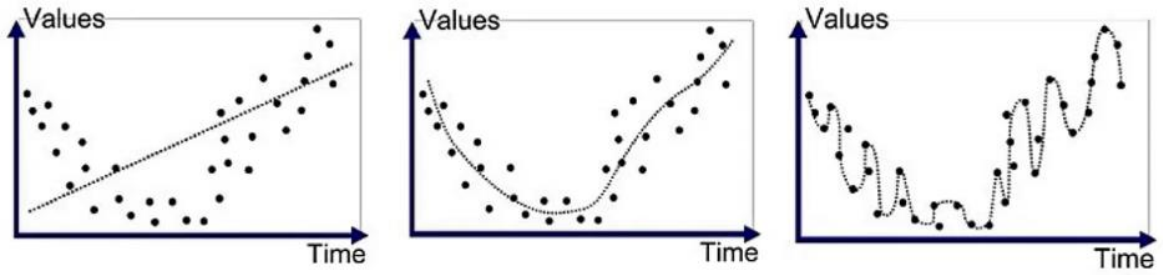

Figura 3.6 *Underfitting* vs. Ajuste correcto vs. *Overfitting*. Fuente: (7 Hidden Layers, s.f.)

Debido al *Overfitting* y al *Underfitting,* además de por la conveniencia de reducir la cantidad de datos suministrados al modelo para mejorar los tiempos de ejecución, es necesario aplicar una selección de atributos que determine qué combinaciones de variables son las óptimas para el desempeño de nuestro modelo. A continuación, se presentan los cuatro métodos empleados en este proyecto para tal fin: la correlación de Pearson, *Mutual Info, Backward Selection* y *LASSO.*

3.2.1.3 Correlación de Pearson

La correlación de Pearson establece una relación entre variables mediante el cálculo de un coeficiente de correlación, la *r de Pearson.* Dicho coeficiente oscila entre los valores de 1 y -1 de forma que:

- ***r = 1:*** Las variables son equivalentes.
- ***0 < r < 1:*** Las variables poseen una relación positiva.
- ***r ≈ 0:*** Las variables no tienen relación alguna.
- ***-1 < r < 0:*** Las variables tienen una relación negativa.
- ***r = -1:*** Las variables son equivalentes e inversas.

Figura 3.7 Distribuciones de variables en función de su *r de Pearson.* Fuente: (Máxima Formación, s.f.).

El cálculo del coeficiente de correlación de Pearson se realiza mediante la siguiente formula:

$$r = \frac{\sigma_{xy}}{\sigma_x . \sigma_y}$$

Donde σ_{xy} es la covarianza entre ambas variables, y σ_x y σ_y son las correspondientes desviaciones típicas.

A la hora de implementar un método de selección de variables basado en la correlación de *Pearson,* es común optar por un procedimiento de filtrado. En él, mediante un proceso iterative, se eligen aquellas variables independientes cuyo coeficiente de correlación con la variable objetivo supere un determinado valor, el cual irá aumentando iteración tras iteración hasta alcanzar un valor límite elegido, siempre inferior a la unidad.

En cada iteración se probará la combinacón de variables obtenida mediante un algoritmo de predicción, calculando su error.

Finalmente, la combinación de variables óptima será aquella cuyo error en la predicción resulte menor.

3.2.1.4 Información Mutua

"La información mutua mide indirectamente la relación entre el conjunto de variables dependientes X y una variable objetivo Y, a través de la medida de la distancia entre la distribución conjunta actual de los datos, *p(x,y),* y la distribución que tendrían si x_i e Y tendrían si fueran independientes, es decir, $p(x_i, y) = p(x_i)p(y)$ si dependiesen linealmente, conocida como distancia de Kullback-Leiber."

(CARDONA M. & VELÁSQUEZ H., 2006).

El valor de la Información Mutua oscilará entre 0 y 1, siendo 0 cuando las variables crezcan de relación alguna, y 1 cuando ocurra justo lo contrario.

La expression para el cálculo de la Información Mutua para el caso continuo es la siguiente:

$$I(X;Y) = \int_x \int_y p(x,y) \cdot log \frac{p(x,y)}{p(x) \cdot p(y)} dxdy$$

El método de selección de variables basado en Información Mutua consistirá pues en determinar los valores de la I.M. para cada variable, procediendo de forma análoga al caso de la correlación de Pearson en lo referente a la elección de la combinación de variables óptima.

3.2.1.5 Eliminación Regresiva

El de Eliminación Regresiva es uno de los algoritmos de selección secuencial de atributos. Los métodos de selección secuencial basan su funcionamiento en la adición o eliminación progresiva de las diferentes variables que componen el set de variables independientes.

En el caso de la eliminación regresiva, el algoritmo comienza con las variables dependientes al completo. El algoritmo, mediante un proceso iterativo, seleccionará cada vez aquellas variable cuya eliminación maximice la optimización del método de predicción elegido. (Raschka, 2014).

3.2.1.6 LASSO

La regresión LASSO es un modelo lineal cuya función es la siguiente:

$$\frac{1}{2 \cdot N_{entrenamieno}} \cdot \sum_{i=1}^{N_{entrenamiento}} (y^{(i)}{}_{real} - y^{(i)}{}_{pred})^2 + \alpha \sum_{j=1}^{n} |a_j|$$

Donde $N_{entrenamiento}$ es el número de datos empleados en el set de entrenamiento, α es un hiperparámetro que ajusta el peso del término de penalidad compuesto por la suma de los coeficientes de correlación de las variables, a_j. (Malato, 2021).

El método minimizará la función, seleccionando automáticamente aquellas variables que sean útiles, rechazando las que no posean valor alguno o sean redundantes.

El uso de la regresión LASSO para la selección de variables se realizará pues, ajustando una recta de regresión a nuestro set de datos y descartando aquellas variables cuyos coeficientes sean cero.

3.2.2 Algoritmos de Predicción

Tras realizar una correcta selección de variables, el siguiente paso para la elaboración de predicciones basadas en algoritmos de aprendizaje automático será, precisamente, la aplicación de dichos métodos.

En el punto anterior se realizó una diferenciación entre los diferentes tipos de aprendizaje existentes en el entorno del *Machine Learning.* Como es natural, dependiendo de la naturaleza del problema será necesario optar por un tipo diferente.

También se hizo mención a los tipos de problemas de predicción existentes, distinguiendo entre problemas de clasificación y problemas de predicción de valores continuos. El objeto de este proyecto es el de predecir los precios futuros del mercado eléctrico diario, por lo que se trata de un problema que involucra variables continuas. Ello hace que los métodos empleados, presentados a continuación, sean métodos enfocados a desempeñar ese tipo de análisis. No obstante, existen métodos de clasificación que pueden ser probados útiles en problemas de predicción de valores contínuos, por lo que se incluirán también en el estudio.

3.2.2.1 Set de entrenamiento y de prueba y cálculo del error

La puesta en marcha de la inmensa mayoría de métodos de aprendizaje automático requiere que se divida el set de datos en dos partes: un set de entrenamiento, y otro de testeo.

El primero será el empleado para "entrenar" al algoritmo, para que en base a sus entradas sea capaz de establecer las relaciones que deduzcan qué ha determinado el valor de las salidas. De esta forma, ante futuras entradas el programa podrá de forma autónoma llegar a proporcionar resultados precisos únicamente a partir de unos imputs recibidos.

El segundo grupo de datos, el de prueba, será el empleado para cuantificar el error. Una vez entrenado el modelo, éste será capaz de elaborar predicciones. Una buena forma de evaluar la precision de las mismas consiste en suministrarle al método en cuestión las entradas del set de testeo, las variables independientes, ontener como resultado las predicciones, las variables dependientes predichas, y compararlas con las variables dependientes reales de ese mismo set.

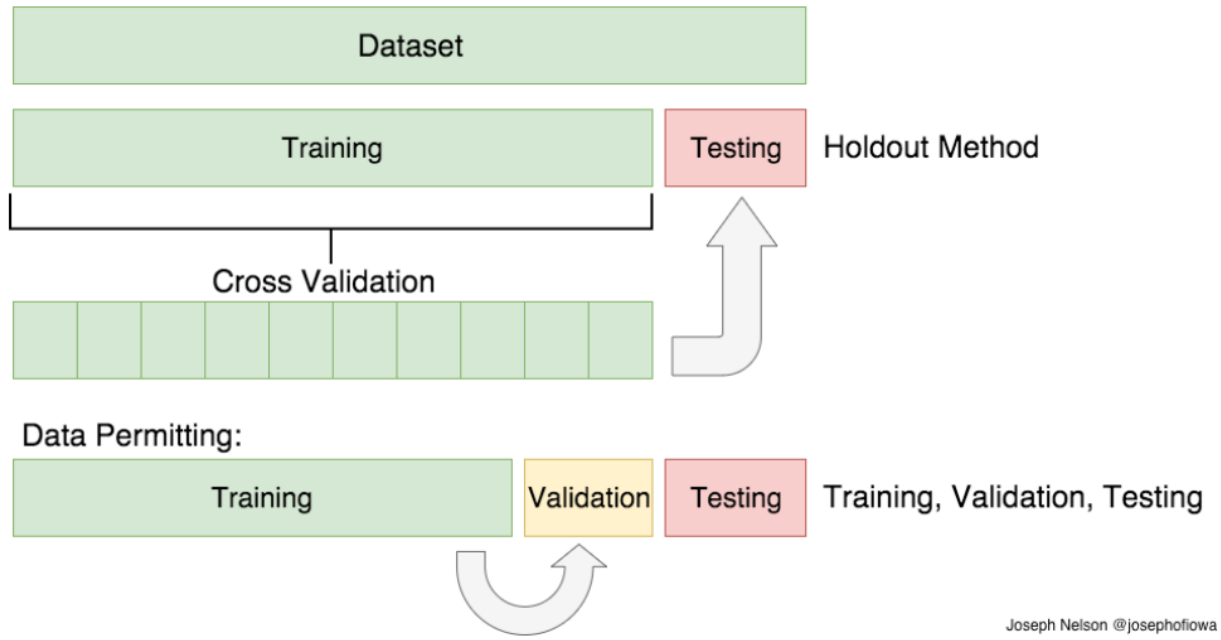

Figura 3.8 Esquema de partición de set de datos original en set de entrenamiento y de testeo.

Fuente: (Bookdown, s.f.)

3.2.2.2 Regresión Lineal

Le regresión lineal es un método que dibuja una recta de regresión a partir del set de datos proporcionado.

La recta de regresión será aquella curva cuya distancia a cada punto del conjunto de datos sea minima.

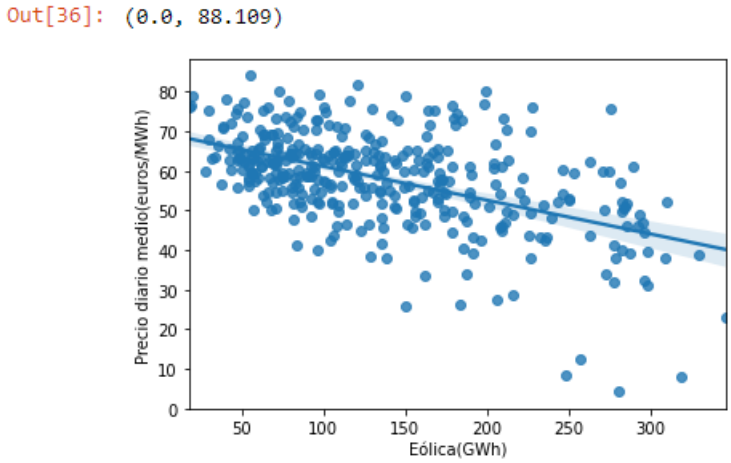

Figura 3.9 Ejemplo de recta de regresión elaborada con Python. Fuente: elaboración propia

La ecuación de la recta de regresión bidimensional es la siguiente:

$$y = mx + b$$

Resulta útil para la comprensión del método explicarlo en su caso bidimensional. Aplicar la regresión lineal en la predicción de valores futuros consistirá en obtener la curva a partir de un conjunto de datos de entrenamiento. Dicha curva aproximará la distribución que sigan dichos datos, la cual, si las muestras de datos son correctas, no será muy diferente de la que otro set de datos pudiera describir. Así, una vez la recta sea trazada, podrá obtenerse un nuevo valor de Y, valor predicho, mediante el corte de un nuevo punto X con dicha curva.

El caso bidimensional es extrapolable a un problema multidisciplinar de n variables cuya ecuación de recta sería la siguiente:

$$y = m_1 \cdot x_1 + m_2 \cdot x_2 + \ldots + m_n \cdot x_n + b$$

3.2.2.3 Árboles de Decisión

Los árboles de decision son algoritmos de aprendizaje supervisado cuyo principio de funcionamiento es el de la división del espacio de variables independientes en regiones diferentes. Es un modelo primordialmente clasificativo, que sin embargo es capaz de operar bien también con valores continuos. (Bookdown, s.f.).

La topología de los árboles de decision se encuentra compuesta de nodos cuya jerarquía disminuye con orden descendente, pudiendo distinguir tres tipos de nodos:

- **Nodos raíz:** en ellos se producen las primeras decisiones en base a la variable más importante del problema.
- **Nodos internos:** continuan las divisiones tras el nodo raíz.
- **Nodos hoja:** son los últimos nodos en la topología e indican la clasificación final.

(Máxima Formación, s.f.).

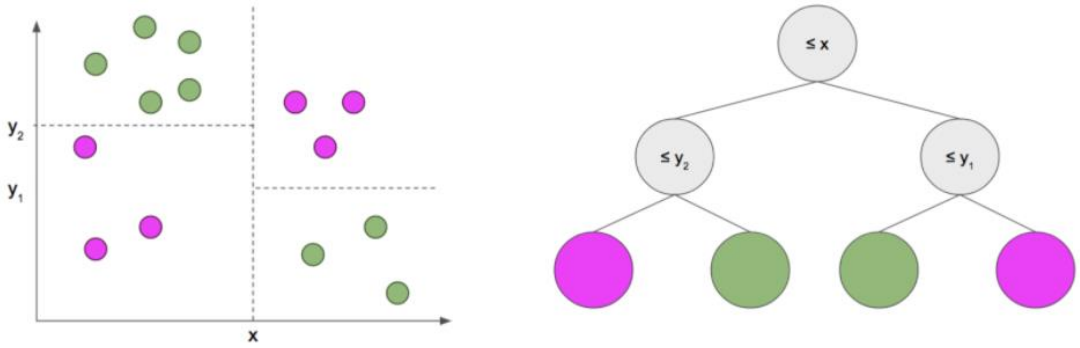

Figura 3.10 Ejemplo de esquema de árbol de decisión. Fuente: (Bookdown, s.f.).

La creación de un árbol de decisión se realiza mediante la aplicación del conocido como algoritmo de *Hunt*, basado en la división de subconjuntos que buscan la separación óptima. Dado un conjunto de datos, se considerará nodo terminal si todos los elementos que lo component pertenecen a la misma clase. De lo contrario, el nodo se dividirá en subconjuntos más pequeños en función de una variable y se repetirá el proceso. (Máxima Formación, s.f.).

En el caso de los árboles de decision regresivos, para la elección de la variable más adecuada para realizar la división, se emplea la suma residual de cuadrados, en inglés *Residual Sum of Squares (RSS)*. El RSS mide la diferencia entre los datos reales y los predichos por el modelo, en este caso con objetivo de minimizarlo.

$$RSS = \sum_{i=1}^{N} (y_i - y'_i)^2$$

3.2.2.4 K – Vecinos Próximos

El método del vecino más cercano consiste en estimar el valor de un dato desconocido en base a las características del dato más próximo, entendiendo por proximidad distancia o similitud. Dicha regla puede ser extendida a más de un dato cercano, convirtiéndose en el método de los K- Vecinos próximos.

(Germán Morales España, 2008).

Pese a ser un método empleado principalmente para la clasificación, el algoritmo de los vecinos próximos puede ser adaptado fácilmente a problemas regresivos. Así, en el caso contínuo, el algoritmo asumirá que todos los datos pertenecen al mismo espacio R_p, y determinará, basándose en la distancia, aquellos *k* datos más cercanos al nuevo dato x_q para aproximar una función *f:* $R^n \rightarrow R$ a partir de los *k* valores ya seleccionados. Dicha función corresponde al promedio de los k valores más cercanos. Considerando el promedio aritmético, la función de aproximación es la siguiente:

$$f(x_q) = \frac{1}{k} \cdot \sum_{i=1}^{k} f(x_i)$$

(Germán Morales España, 2008).

3.2.2.5 Máquinas de Soporte Vectorial

Las máquinas de soporte vactorial (SVM) son algoritmos tanto de clasificación como de regresión cuya versatilidad en las aplicaciones de clasificación han hecho de ellas uno de los mejores métodos para ese cometido. Para comprender el funcionamiento del método basta con entender cuatro conceptos: los hiperplanos de separación, el hiperplano óptimo, el márgen suave y la función kernel o núcleo. (Ochoa, 2020).

Se entiende por hiperplano en un espacio p-dimensional a un subespacio plano y afín, es decir, que no tiene porqué pasar por el origen, de dimensiones p-1. (Ochoa, 2020).

Las máquinas de soporte vactorial aprenden con gran facilidad para aprender a partir de los datos de un set de entrenamiento. El objetivo del algoritmo será el de encontrar la función capaz de separar los datos en dos hiperplanos, y que sea capaz de pronosticar correctamente nuevas muestras. (Ochoa, 2020).

Las SMV hallarán el hiperplano óptimo, aquel que maximice la distancia entre él y el dato más cercano de cada clase. (Ochoa, 2020).

El hiperplano de separación óptimo (HSO), cuya ecuación es $w \cdot x + b = 0$, estará determinado, pues, por el márgen máximo de separación entre las dos clases. Dicho márgen será la distancia entre los planos paralelos $w \cdot x + b = -1$ y $w \cdot x + b = 1$ con el HSO. *W* es el vector de pesos, el cual contiene la ponderación de cada atributo, es decir, el peso que cada variable tiene en la regresión. En resumen, el HSO será aquel que separe a las dos clases y se encuentre a la misma distancia de ambas. El problema de las SVM se resume, pues, a encontrar el mejor HSO posible. (Ochoa, 2020).

En algunas situaciones los datos no son linearmente separables, dificultando la resolución del problema. Ante esas situaciones existen dos maneras de proceder: el márgen suave y los kernels. (Ochoa, 2020).

En el caso del márgen suave, un parámetro de regularización, *C*, es añadido al hiperplano. El valor de *C* detarminará cuántas observaciones incorrectas pueden ser ubicada por el algoritmo en el espacio incorrecto, por lo que un valor reducido de *C* supondrá una mayor precision del modelo. (Ochoa, 2020).

Abordar el problema no lineal mediante kernels implica pasar de un espacio de *n* dimensiones a otro de *h* dimensiones, siendo h mayor que n. El objetivo será hallar un HSO lineal en un espacio de mayores dimensiones que el original, en el cual el hiperplano de separación óptimo no sería lineal. (Ochoa, 2020).

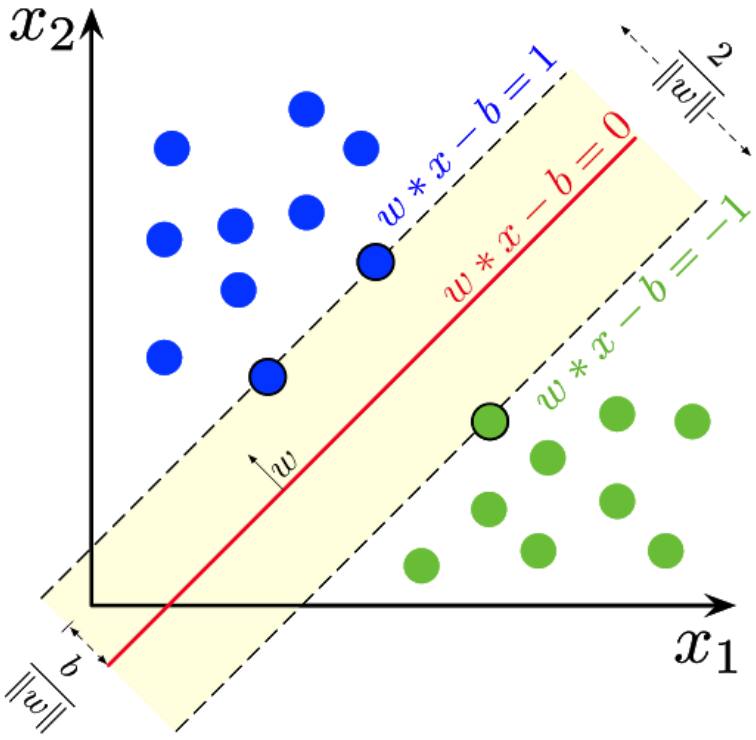

Figura 3.11 Representación del HSO con sus márgenes correspondientes. Fuente: (Barrán, 2019).

3.2.2.6 Redes Neuronales

Una red neuronal es un modelo simplificado de la forma de profesar la información del cerebro humano. Funciona simultaneando múltiples unidades individulales de procesamiento que recuerdan a las neuronas cerebrales. Es, por lo tanto, un algoritmo de aprendizaje automatico. (IBM, 2020).

Las unidades de procesamiento (neuronas) se organizan en capas, existiendo una capa de entrada, una o varias capas ocultas, y una capa de salida. Dichas unidades se encuentran conectadas entre sí mediante el uso de ponderaciones. Así, los datos son presentados en la capa de entrada de la red. Éstos van circulando desde cada neurona a la neurona de la capa siguiente hasta llegar a la capa de salida. (IBM, 2020).

El proceso de aprendizaje de la red consiste en examinar los registros individuales, generando una predicción para cada registro, y ajustando las ponderaciones de casa neurona cuando ésta realiza una predicción incorrecta. Es un proceso iterativo que se repite hasta que se alcance uno de los criterios de parada. (IBM, 2020).

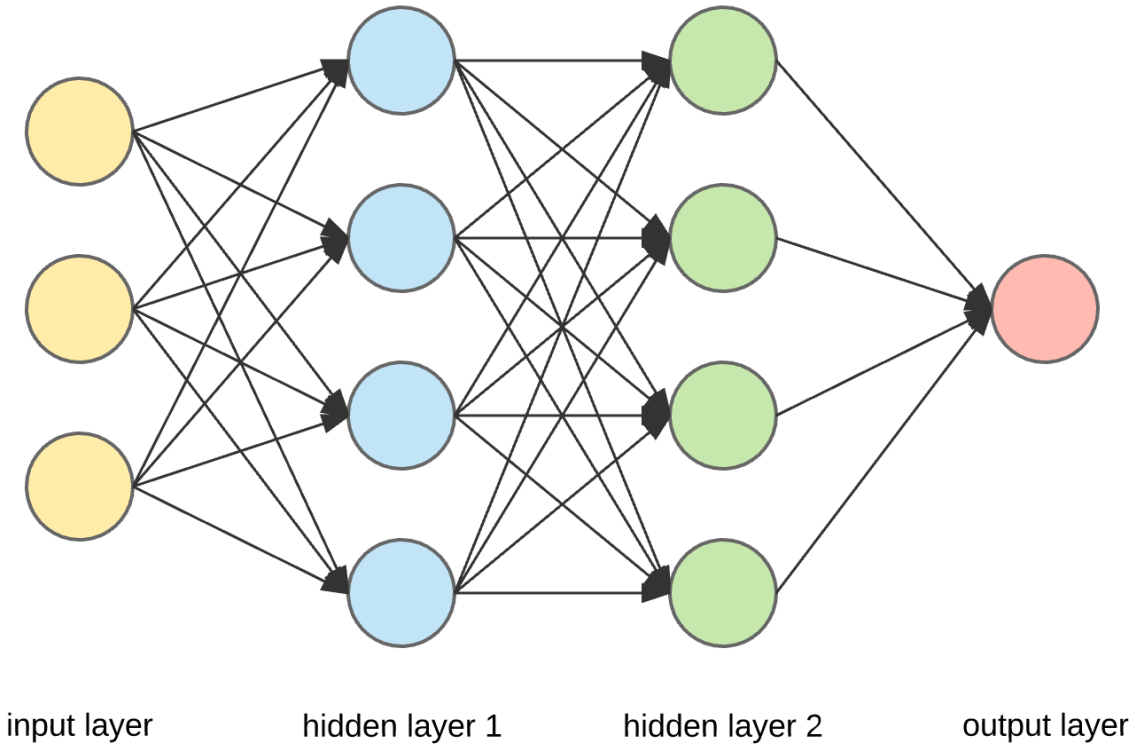

Figura 3.12 Esquema de red neuronal con dos capas ocultas. Fuente: (Gong, 2019).

3.2.2.7 Errores

A la hora de determinar la precision de un método es fundamental contar con herramientas que cuantifiquen dicho error.

Existen diferentes formas de cuantificar la precision de los métodos de predicción, encontrándose varias de ellas includas distintas librerías de *Python.* Para evaluar los errores cometidos por los diferentes métodos de selección de variables empleados en este proyecto se han elegido el coeficiente de proporción, *r cuadrado*, y la raíz cuadrada del error cuadrático medio, *RMSE*:

- ***r cuadrado***:

$$r^2 = \frac{\sum_{t=1}^{T}(\hat{Y}_t - \bar{Y})^2}{\sum_{t=1}^{T}(Y_t - \bar{Y})^2}$$

Donde $\widehat{Y_t}$ son las diferentes estimaciones realizadas por el algoritmo, $\bar{Y}$ es la media de las variables dependientes real, e Y_t son las variables dependientes reales. En nuestro caso, $\widehat{Y_t}$ serían los precios estimados por la máquina, $\bar{Y}$ la media de los precios reales del set de testeo, e Y_t serían los precios reales del set de prueba.

El coeficiente de proporción podría entenderse como un tanto por uno que determina la precision del método en cuestión. De esa forma, una r^2 de 0,81 podría entenderse como que el método tiene una precision del 81%. (Economipedia, 2017).

- ***RMSE:***

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{M}(\widehat{Y_i} - Y_i)^2}{M}}$$

Donde $\widehat{Y_i}$ e Y_i se corresponden con las del coefidiente de proporción. Al tratarse de un error, la precision de un método será mayor cuanto menor *RMSE* posea.

Los estimadores elegidos para determinar la exactitud de los algoritmos de predicción difieren de los empleados para los de selección de atributos. En este caso se optó por emplear el error cuadrático medio, *MSE,* correspondiente al cuadrado del *RMSE*, y el error medio absolute, *MAE.*

- ***MSE:***

$$\mathrm{MSE} = \frac{\sum_{i=1}^{M}(\widehat{Y_i} - Y_i)^2}{M}$$

- ***MAE:***

$$\mathrm{MAE} = \frac{\sum_{i=1}^{M}\left|Y_i - \widehat{Y_i}\right|}{M}$$

El *MAE* se traduce en la diferencia media entre los precios reales del set de testeo y los precios predichos por el método en cuestión, por lo que aporta información sobre en cuándos euros de media se equivoca el algoritmo a la hora de realizar una predicción.

4 ESTUDIO COMPARATIVO DE MÉTODOS DE PREDICCIÓN

Locura es hacer lo mismo una y otra vez
esperando obtener resultados diferentes

- Rita Mae Brown -

En el punto anterior se describieron los diferentes métodos de selección de variables y de predicción empleados para el estudio realizado en este proyecto. En este apartado se procede, pues, a explicar dicho anáisis. Para ello se abordarán todos los aspectos de este, la selección de la información, su tratamiento y selección de variables, la comparativa entre modelos y, por último, los resultados y las conclusiones.

4.1 Selección de Datos

La selección de datos para el desarrollo del proyecto se ha limitado exclusivamente al ámbito del sector eléctrico, quedando excluídas otras variables como las meteorológicas, el precio de materias primas como el gas o el petróleo, o aspectos macroeconómicos como los mercados financieros o el PIB nacional.

Las variables seleccionadas para el desarrollo del estudio, todas ellas muestras horarias, son el precio del megavatio hora en euros, los datos de generación de las diferentes tecnologías que conforman el mix eléctrico español (carbon, ciclo combinado, hidráulica, eólica, nuclear, solar térmica, solar fotovoltaica, cogeneración y otros), la demanda eléctrica nacional, y los intercambios energéticos con Baleares e internacionales.

4.2 Tratamiento de Datos

Los datos fueron obtenidos de la web oficial de *Red Eléctrica de España (REE),* siendo elegido el periodo transcurrido desde el día 1 de enero de 2021 hasta el día 3 de mayo de 2021. Cada variable fue obtenida mediante la descarga de un fichero de excel independiente, los cuales fueron agrupados a posteriori en un único fichero que reuniese toda la información con el objetivo de trabajar desde un único fichero con *Python.* Dicho fichero contaba con un total de 2951 filas, una por día, y 13 columnas, 12 variables y 1 correspondiente a la fecha, la cual fue eliminada debido a su irrelevancia para el estudio y su formato incompatible.

Out[6]:

	Fecha	Precio	Demanda	Carbón	Ciclo Combinado	Cogeneración y Resto	Eólica	Hidráulica	Nuclear	Solar FV	Solar Térmica	Intercambios Baleares	Intercambios Internacionales
0	2021-01-01T00:00:00+01:00	54.56	24722.833	260.667	2519.167	2400.333	6883.667	6225.667	7118.500	8.667	0.000	-109.000	-1116.167
1	2021-01-01T01:00:00+01:00	52.80	23571.500	261.000	2523.500	2366.167	6978.500	5776.333	7116.833	8.167	0.000	-100.000	-1905.500
2	2021-01-01T02:00:00+01:00	49.95	21491.000	258.167	2133.167	2357.667	7229.333	4715.000	7116.500	8.167	0.000	-100.000	-2771.667
3	2021-01-01T03:00:00+01:00	45.22	19709.167	258.500	1775.500	2353.333	7878.000	3081.500	7117.500	7.667	0.000	-100.000	-3228.333
4	2021-01-01T04:00:00+01:00	42.91	18791.667	258.000	1730.833	2355.500	8718.667	2093.000	7118.167	8.000	0.000	-100.000	-3935.667
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2946	2021-05-03T19:00:00+02:00	87.07	27627.333	423.833	3810.833	3517.000	3862.000	3710.500	6953.667	2207.833	725.000	-176.500	2005.333
2947	2021-05-03T20:00:00+02:00	91.33	28845.500	423.667	4187.000	3532.833	3720.000	4853.500	6986.333	711.833	581.167	-227.500	3456.833
2948	2021-05-03T21:00:00+02:00	91.69	30452.167	423.833	4517.000	3536.333	3294.000	6365.833	7019.833	26.500	365.333	-232.667	4547.333
2949	2021-05-03T22:00:00+02:00	88.01	28735.000	424.167	4399.667	3532.833	2901.667	5550.500	7044.667	6.333	299.667	-183.000	4162.333
2950	2021-05-03T23:00:00+02:00	86.33	26216.500	415.833	4142.667	3508.667	2888.333	3954.833	7046.667	5.667	296.167	-109.167	3481.167

2951 rows × 13 columns

Figura 4.1 Captura del data frame compuesto por los datos obtenidos de REE. Fuente: elaboración propia

Out[7]:

	Precio	Demanda	Carbón	Ciclo Combinado	Cogeneración y Resto	Eólica	Hidráulica	Nuclear	Solar FV	Solar Térmica	Intercambios Baleares	Intercambios Internacionales
0	54.56	24722.833	260.667	2519.167	2400.333	6883.667	6225.667	7118.500	8.667	0.000	-109.000	-1116.167
1	52.80	23571.500	261.000	2523.500	2366.167	6978.500	5776.333	7116.833	8.167	0.000	-100.000	-1905.500
2	49.95	21491.000	258.167	2133.167	2357.667	7229.333	4715.000	7116.500	8.167	0.000	-100.000	-2771.667
3	45.22	19709.167	258.500	1775.500	2353.333	7878.000	3081.500	7117.500	7.667	0.000	-100.000	-3228.333
4	42.91	18791.667	258.000	1730.833	2355.500	8718.667	2093.000	7118.167	8.000	0.000	-100.000	-3935.667
...	...	...	...	...	...	...	...	...	...	...	...	...
2946	87.07	27627.333	423.833	3810.833	3517.000	3862.000	3710.500	6953.667	2207.833	725.000	-176.500	2005.333
2947	91.33	28845.500	423.667	4187.000	3532.833	3720.000	4853.500	6986.333	711.833	581.167	-227.500	3456.833
2948	91.69	30452.167	423.833	4517.000	3536.333	3294.000	6365.833	7019.833	26.500	365.333	-232.667	4547.333
2949	88.01	28735.000	424.167	4399.667	3532.833	2901.667	5550.500	7044.667	6.333	299.667	-183.000	4162.333
2950	86.33	26216.500	415.833	4142.667	3508.667	2888.333	3954.833	7046.667	5.667	296.167	-109.167	3481.167

2951 rows × 12 columns

Figura 4.2 Data frame tras eliminar columna de fechas. Fuente: elaboración propia

4.3 Selección de Variables

Para la selección de variables se emplearon los métodos mencionados en el punto anterior: correlación de *Pearson*, información mutua, selección recursiva y *LASSO*.

4.3.1 Correlación de *Pearson*

El primer paso tomado para aplicar la selección por filtrado por correlación fue el de calcular la matriz de coeficientes de Pearson, la cual fue obtenida en forma de mapa de calor.

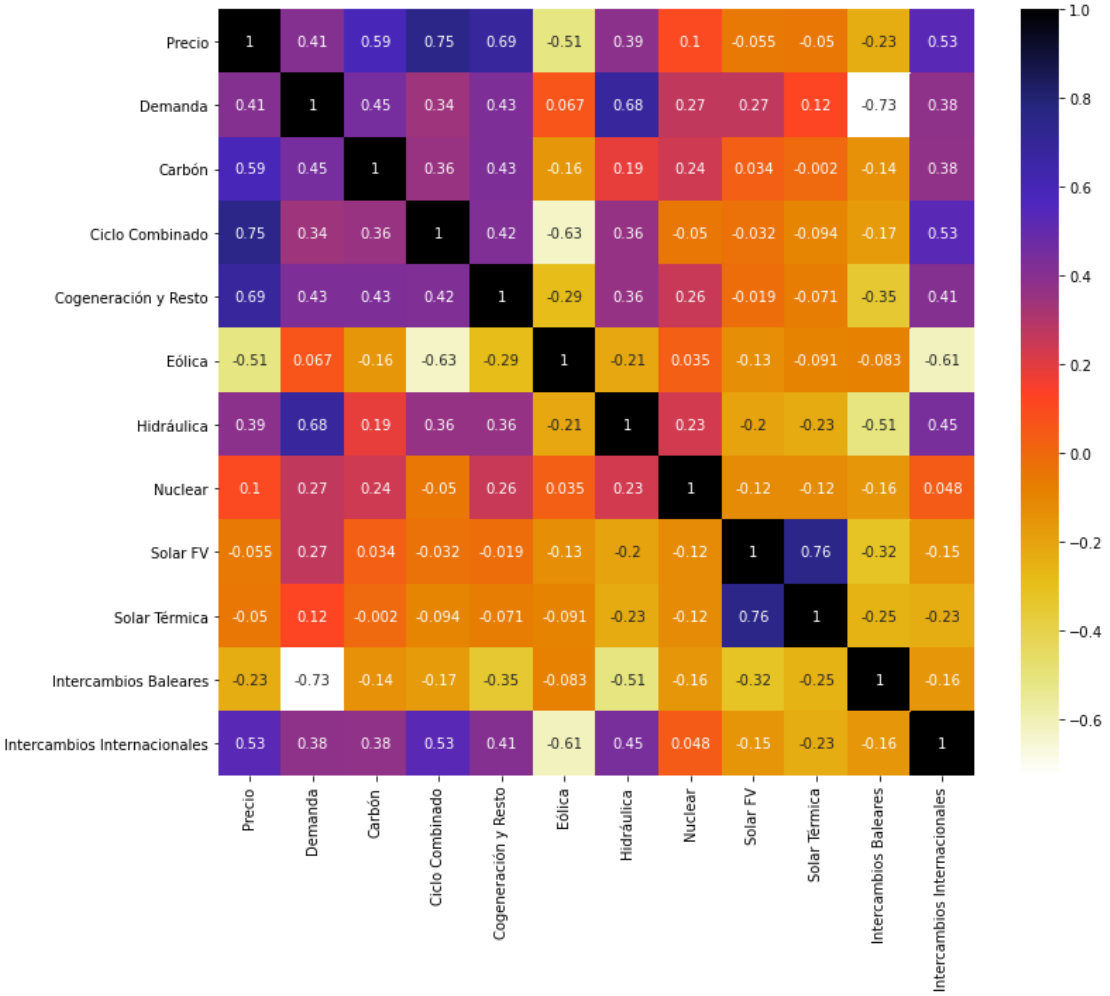

Figura 4.3 Mapa de calor de las correlaciones de Pearson de cada variable. Fuente: elaboración propia

Tras la obtención de los oeficientes de correlacion de todas las variables, se proceció a realizar el proceso iterativo mencionado en el apartado anterior, determinando qué combinación de variables ofrecía un resultado más preciso. Como método de predicción empleado para testear las combinaciones se optó por emplear el método de validación cruzada.

La combinación óptima resultante fue aquella que incluía las generaciones de carbón, ciclo combinado, cogeneración y resto, eólica e intercambios internacionales, con una *r cuadrado* de 0,81 y un *RMSE* de 10,52.

4.3.2 Información mutua

El siguiente algoritmo implementado para la selección de atributos fue el método de información mutua. En primer lugar, se calcularon los pesos correspondientes a cada variable, obteniéndose los siguientes resultados:

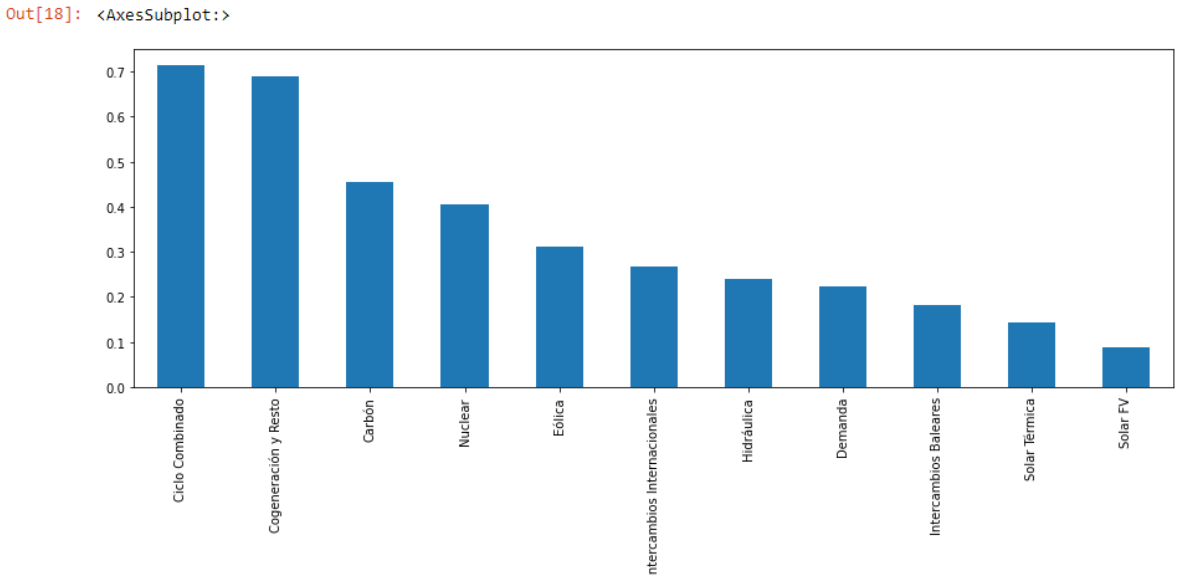

Figura 4.4 Ponderaciones de cada atributo respecto al precio del MWh. Fuente: elaboración propia.

Una vez obtenidos los resultados, se optó por comparar los errores obtenidos si se procedía a la eliminación una o ambas tecnologías solares, o si se ampliaba la criba a los intercambios con las Islas Baleares.

En el primer caso, eliminando todas las tecnologías solares se obtuvo una *r cuadrado* de 0,779 y un *RMSE* de 11,385. En cambio, al eliminar únicamente la generación solar fotovoltaica la precisión bajó hasta una *r cuadrado* de 0,762 y un *RMSE* de 11,812. Por último, al excluir además los intercambios con Baleares, la *r cuadrado* obtenida fue de 0,777, y el *RMSE* de 11,452.

La mejor combinación aplicando el criterio de información mutua, pues, resultó ser el que exluía las tecnologías de generación solar.

4.3.3 Eliminación Recursiva

En tercer lugar se aplicó el método de eliminación recursiva. Dicho método arroja directamente la combinación de variables óptima, por lo que tras ejecutar el código correspondiente se obtuvo el resultado de inmediato. La combinación de variables óptima resultó ser aquella que englogaba a la demanda, las generaciones de carbón, ciclo combinado y cogeneración y resto, y los intercambios energéticos con Baleares, con una *r cuadrado* de 0,822, y un *RMSE* de 10,238.

4.3.4 LASSO

Por ultimo, se realizó una selección de variables empleando el método *LASSO*. El método arrojó una combinación de variables óptima coincidente con la obtenida mediante el método de correlación de *Pearson.*

4.3.5 Tabla comparativa de Resultados

A continuación se adjunta la tabla comparativa obtenida:

Tabla 4-1. Errores obtenidos para cada método de selección de atributos.

Método	*RMSE*	*R cuadrado*
Original	11,38	0,779
Pearson	10,52	0,81
Info mutua	11,81	0,79
Sel. Regresiva	10,24	0,82

4.4 Comparativa entre Modelos

Tras la obtención de las tres combinaciones óptimas de atributos, se procedió a ejecutar todos los métodos de predicción objeto de estudio con cada una de las combinaciones de variables obtenidas en el apartado anterior. Por ello, además del método, se observó con qué set de variables se alcanzó el resultado más preciso, pues la exactitud de las combinaciones de variables puede verse alterada en función del algoritmo empleado.

Todas las combinaciones de variables seleccionadas fueron divididas en un set de prueba y en otro de testeo.

4.4.1 Regresión Lineal

La tabla de resultados obtenidos tras aplicar el método de regresión lineal es la siguiente:

Tabla 4-2. Errores obtenidos mediante el método de regresión lineal

Combinación	*MAE*	*MSE*
Pearson	8,4377	112,684
Info mutua	8,7085	132,7027
Sel. Regresiva	8,222	126,2505

4.4.2 Árboles de Decisión

La tabla de resultados obtenidos tras aplicar el método de árboles de regresión es la siguiente:

Tabla 4-3. Errores obtenidos mediante el método de árboles de regresión

Combinación	*MAE*	*MSE*
Pearson	6,3378	81,6769
Info mutua	6,3558	83,3875
Sel. Regresiva	6,6321	89,7175

4.4.3 K Vecinos Próximos

Para el caso aplicado de la predicción mediante el método de los K vecinos próximos fue necesario un paso previo a la ejecución de la predicción. El método precisa el número k de vecinos con los que realizar la predicción, por lo que éste hubo de ser determinado. Para ello, se hallaron los errores en la precisión desde k = 1 hasta k = 50, obteniendo las siguientes gráficas para los tres casos estudiados:

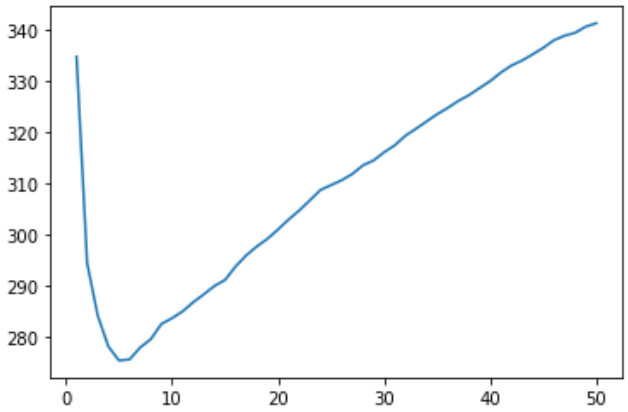

Figura 4.5. Error cuadrático medio (Y) frente a número de vecinos (X). Primera combinación. Fuente: elaboración propia

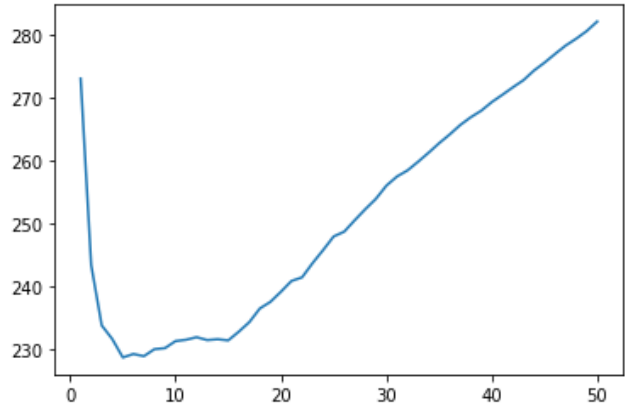

Figura 4.6. Error cuadrático medio (Y) frente a número de vecinos (X). Segunda combinación. Fuente: elaboración propia

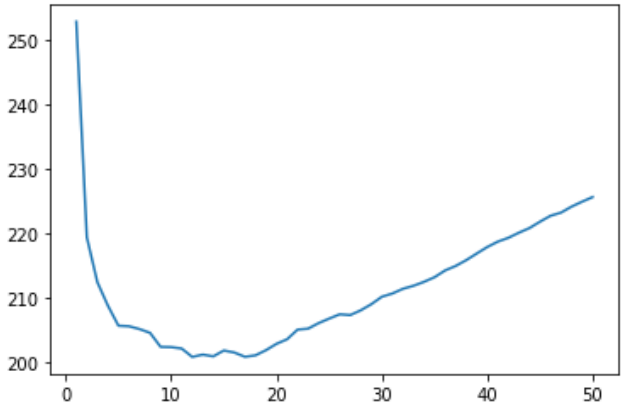

Figura 4.7. Error cuadrático medio (Y) frente a número de vecinos (X). Tercera combinación. Fuente: elaboración propia

El número de vacions óptimo, *k*, resultó ser 5 para el primer y segundo caso, y 13 para el tercero.

La tabla de resultados obtenidos tras aplicar el método de los K vecinos próximos es la siguiente:

Tabla 4-4. Errores obtenidos mediante el método de los K vecinos próximos

Combinación	*MAE*	*MSE*
Pearson	10,2039	275,5643
Info mutua	9,5423	229,1023
Sel. Regresiva	8,723	202,8554

4.4.4 Máquinas de Soporte Vectorial

La tabla de resultados obtenidos tras aplicar el método de los K vecinos próximos es la siguiente:

Tabla 4-5. Errores obtenidos mediante el método de las máquinas de soporte vectorial

Combinación	*MAE*	*MSE*
Pearson	12,6276	252,0135
Info mutua	13,5411	304,7043
Sel. Regresiva	12,984	286,0388

4.4.5 Redes Neuronales

Al igual que en el caso del método de los K vecinos próximos, las redes neuronales requieren cierta configuración previa para su correcto funcionamiento.

El ajuste de una red neuronal se realiza mediante la alteración de dos valores principalmente: el primero, el número de capas de neuronas ocultas. El segundo, la cantidad de neuronas por capa.

En nuestro caso, todas las redes neuronales fueron construidas con dos capas ocultas de neuronas, y cien neuronas por capa. Dicho ajuste se realizó mediante el ensayo y el error, visualizando las curvas de ajuste del *MAE* y el *MSE* de las redes, adjuntadas a continuación:

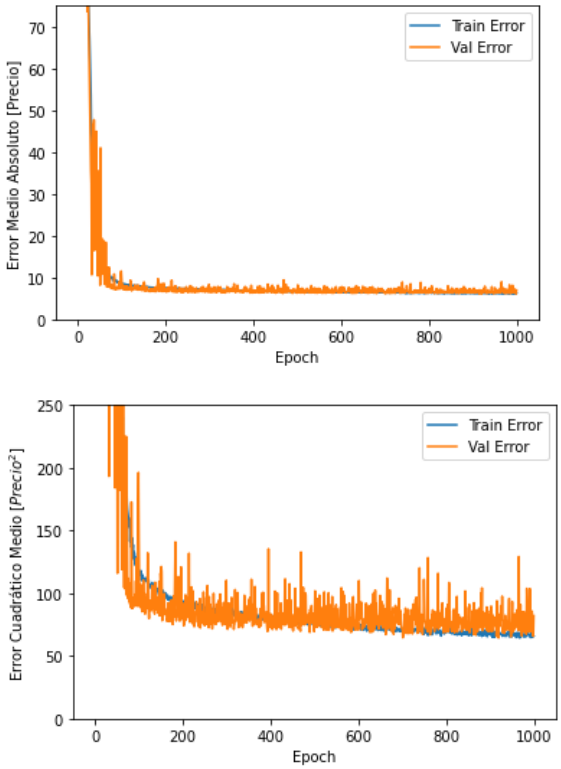

Figura 4.8. Primera combinación. Arriba: MAE frente a ciclos de ejecución (Epoch). Abajo: MSE frente a ciclos de ejecución (Epoch). Fuente: elaboración propia.

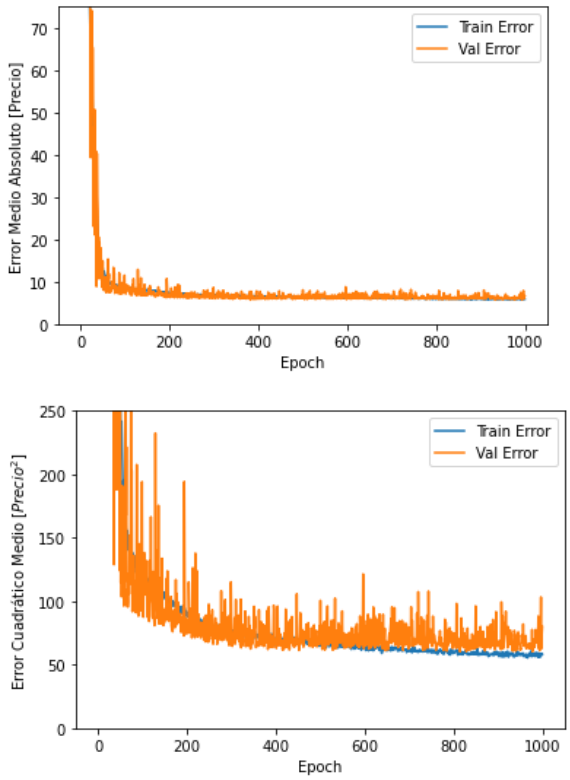

Figura 4.9. Segunda combinación. Arriba: MAE frente a ciclos de ejecución (Epoch). Abajo: MSE frente a ciclos de ejecución (Epoch). Fuente: elaboración propia.

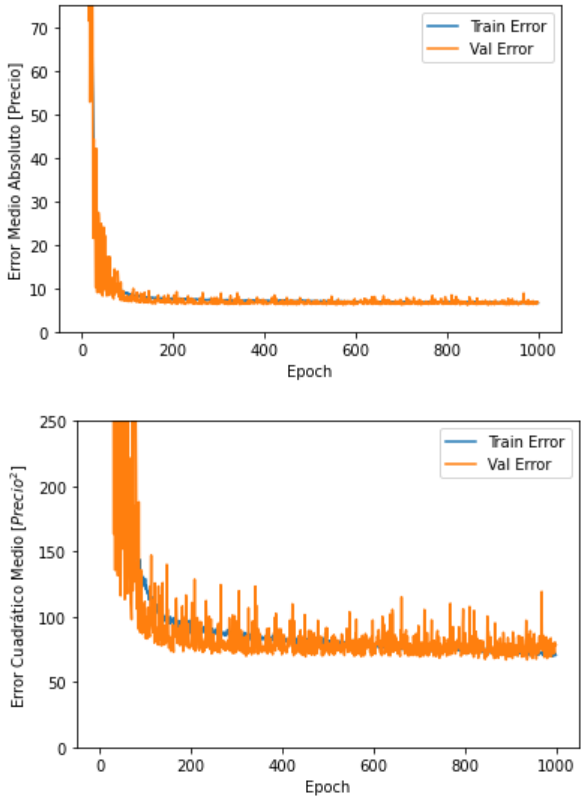

Figura 4.10. Primera combinación. Arriba: MAE frente a ciclos de ejecución (Epoch). Abajo: MSE frente a ciclos de ejecución (Epoch). Fuente: elaboración propia.

La tabla de resultados obtenidos tras aplicar el método de redes neuronales es la siguiente:

Combinación	*MAE*	*MSE*
Pearson	7,00	83,00
Info mutua	6,80	77,00
Sel. Regresiva	7,00	80,00

4.5 Conclusiones

El objetivo fundamental del proyecto plasmado en este documento era el de determinar qué método de aprendizaje automático era capaz de realizar la predicción del precio diario del megavatio hora más ajustada. Para ello, se realizaron varios pasos.

En primer lugar, se eligieron los datos a emplear que servirían como base para el Desarrollo de las futuras predicciones.

Tras ello, se procedió a seleccionar las variables más significativas para realizar dicha predicción. Se emplearon cuatro métodos diferentes para ello, arrojando tres combinaciones de variables diferentes, pues dos de los métodos proporcionaron el mismo resultado.

Finalmente, se elaboraron los diferentes modelos de machine learning a partir de las combinaciones de variables elegidas, probándose cada uno para todas ellas.

A priori, como se desarrolló anteriormente, los métodos regresivos suelen obtener buenos resultados cuando se les emplea para la predicción de valores contínuos como los precios, por lo que era de esperar un buen desempeño del método de regresión lineal. El resultado, sin embargo, resultó ser el tercero mejor de los cinco métodos probados.

El amplio rango de precios existente en los datos históricos dificultó que las máquinas vectoriales realizasen una buena predicción debido a su algoritmo de clasificación binario. Como resultado, las SVM han obtenido las peores predicciones del conjunto de métodos elegidos.

Pese a ser un método predominantemente clasificativo, el de los K vecinos próximos ha obtenido una predicción aceptable superando a las SVM, aunque aun así no logre superar al resto de algoritmos.

Las redes neuronales demostraron ser un poderoso mecanismo para la predicción de precios. Pese a contar únicamente con dos capas ocultas, el error en la predicción no superó los 7 euros por megavatio hora. Pese a sus ventajas, el coste computacional requerido es considerable. Superó con diferencia cualquier otro método empleado. Con todo ello, las redes neuronales constituyen un método con enorme potencial para la predicción de precios, aunque el uso de redes de gran tamaño pueda implicar la necesidad de grandes cantidades de recursos de procesamiento.

Finalmente, el método que resultó ser más preciso para la predicción de los precios del mercado eléctrico diario fue el de los árboles de decision, probando ser súmamente versátil dada su calidad también como método de clasificación. El error mínimo obtenido resultó ser de 6 euros y 34 céntimos el megavatio hora.

	Pearson	Información Mutua	Selección Regresiva
Lineal	*MAE: 8,4377*	*MAE: 8,7085*	*MAE: 8,222*
	MSE: 112,684	*MSE: 132,7027*	*MSE: 126,2505*
Ridge	*MAE: 8,4377*	*MAE: 8,7085*	*MAE: 8,222*
	MSE: 112,684	*MSE: 132,7027*	*MSE: 126,2505*
Lasso	*MAE: 8,4377*	*MAE: 8,7085*	*MAE: 8,222*
	MSE: 112,684	*MSE: 132,7027*	*MSE: 126,2505*
A. Decisión	*MAE: 6,3378*	*MAE: 6,3558*	*MAE: 6,6321*
	MSE: 81,6769	*MSE: 83,3875*	*MSE: 89,7175*
K. Vecinos	*MAE: 9,2039*	*MAE: 8,5423*	*MAE: 7,723*
	MSE: 141,3625	*MSE: 120,3639*	*MSE: 102,8554*

Tabla 4-6 Comparativa de resultados finales. Fuente: elaboración propia

4.6 Consideraciones Futuras

El estudio realizado en este proyecto realiza una comparativa entre diferentes métodos de predicción de precios aplicados al mercado eléctrico, arrojando un resultado que esclarece cuáles son más indicados para tal cometido. No obstante, dicha investigación podría ampliarse pudiendo elegir entre dos aspectos a optimizer.

En primer lugar, sería interesante ampliar la variedad de datos empleados para la construcción de los modelos. Una Buena forma de hacerlo sería la adición de datos meteorológicos como las temperaturas, la pluviometría o la irradiación solar. Además, podría ser de utilidad estudiar cómo afectan variables macroeconómicas como el PIB, el precio de las materias primas o diferentes sucesos geopolíticos al precio del kilovatio hora.

Tras ampliar el elenco de datos con los que entrenar los diferentes modelos, podría ponerse el foco en la optimización de los métodos que mejores predicciones han arrojado en este estudio. Así, podrían ajustarse los parámetros del método de los vecinos próximos, y especialmente ampliar y optimizar las redes neuronales empleadas añadiendo más neuronas por capa, más capas de neuronas, o variando los ciclos de ejecución de las mismas.

5 APÉNDICES

5.1 Código empleado

5.1.1 Tratamiento Previo de Datos

```
import pandas as pd
datos = pd.read_excel(r'C:\Users\Eugenio\Desktop\Carrera\Cuarto\Segundo cuatri\TFG\Datos_2021.xlsx')
datos
datos.drop('Fecha', axis=1, inplace=True)
datos
X = datos.drop("Precio", axis=1)
y = datos["Precio"]
```

5.1.2 Selección de atributos

```
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import KFold
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import cross_val_predict
from sklearn.linear_model import LinearRegression
from math import sqrt

modelo = LinearRegression()
modelo.fit(X, y)
y_pred = modelo.predict(X)
#cv = KFold(n_splits=10, random_state=None, shuffle=False)
#classifier_pipeline = make_pipeline(StandardScaler(), KNeighborsRegressor(n_neighbors=10))
#y_pred = cross_val_predict(classifier_pipeline, X, y, cv=cv)
RMSE_original = sqrt(mean_squared_error(y,y_pred))
R2_original = r2_score(y,y_pred)
print("RMSE: " + str(round(sqrt(mean_squared_error(y,y_pred)),2)))
print("R_squared: " + str(round(r2_score(y,y_pred),2)))
R2_original
```

5.1.2.1 Correlación de Pearson

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
plt.figure(figsize=(12,10))
cor = datos.corr()
sns.heatmap(cor, annot=True, cmap=plt.cm.CMRmap_r)
plt.show()
abs(datos.corr()["Precio"][abs(datos.corr()["Precio"])>0.5].drop('Precio')).index.tolist()
vals = [0.1,0.2,0.3,0.4,0.5,0.6,0.7]
for val in vals:
    features = abs(datos.corr()["Precio"][abs(datos.corr()["Precio"])>val].drop('Precio')).index.tolist()

    X = datos.drop(columns='Precio')
    X=X[features]

    print(features)
    y_pred = cross_val_predict(classifier_pipeline, X, y, cv=cv)
    print("RMSE: " + str(round(sqrt(mean_squared_error(y,y_pred)),2)))
    print("R_squared: " + str(round(r2_score(y,y_pred),2)))
```

5.1.2.2 Información Mutua

```
from sklearn.feature_selection import mutual_info_regression

X = datos.drop("Precio", axis=1)
y = datos["Precio"]
mutual_info = mutual_info_regression(X, y)
mutual_info
mutual_info = pd.Series(mutual_info)
mutual_info.index = X.columns
mutual_info.sort_values(ascending=False)
mutual_info.sort_values(ascending=False).plot.bar(figsize=(15,5))
X = datos.drop(columns='Precio')
y = datos['Precio']
#y_pred = cross_val_predict(classifier_pipeline, X, y, cv=cv)



print("RMSE: " + str(round(sqrt(mean_squared_error(y,y_pred)),3)))
print("R_squared: " + str(round(r2_score(y,y_pred),3)))
```

5.1.2.3 Selección Regresiva

```
from mlxtend.feature_selection import SequentialFeatureSelector as SFS
sfs1 = SFS(classifier_pipeline,
        k_features=1,
        forward=False,
        scoring='neg_mean_squared_error',
        cv=cv)

X = datos.drop(columns='Precio')

sfs1.fit(X,y)

sfs1.subsets_
X = datos.drop(columns='Precio')[['Demanda', 'Carbón','Ciclo Combinado','Cogeneración y Resto','Intercambios Baleares']]
y = datos['Precio']
y_pred = cross_val_predict(classifier_pipeline, X, y, cv=cv)
RMSE_bw = sqrt(mean_squared_error(y,y_pred))
R2_bw = r2_score(y,y_pred)
print("RMSE: " + str(round(sqrt(mean_squared_error(y,y_pred)),3)))
print("R_squared: " + str(round(r2_score(y,y_pred),3)))
```

5.1.2.4 LASSO

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import scipy as sp
import scipy.stats as stats
import seaborn as sns
import statsmodels.api as sm
import warnings
from mpl_toolkits import mplot3d
from pylab import rcParams
from scipy import stats
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split, GridSearchCV
j = sns.jointplot("Precio", "Ciclo Combinado", data = datos, kind = 'reg')
j.annotate(stats.pearsonr)
```

```
plt.show()
X = datos.drop("Precio", axis=1)
y = datos["Precio"]
lasso = Lasso()
params = {"alpha" : [1e-15, 1e-10, 1e-8, 1e-4, 1e-3, 1e-2, 1, 1e1,
                1e2, 1e3, 1e4, 1e5, 1e6, 1e7]}
lasso_regressor = GridSearchCV(lasso, params,
                        scoring="neg_mean_squared_error",
                        cv=5)
lasso_regressor.fit(X, y)
lasso_regressor.best_score_
lasso_regressor.best_estimator_
lasso_best = lasso_regressor.best_estimator_
lasso_best.fit(X, y)
coef = pd.Series(lasso_best.coef_,list(X.columns))
coef.plot(kind='bar', title='Model Coefficients')
X = datos.drop(columns='Precio')[['Demanda', 'Carbón', 'Ciclo Combinado', 'Cogeneración y Resto', 'Intercambios Internacionales', 'Eólica']]
y = datos['Precio']
y_pred = cross_val_predict(classifier_pipeline, X, y, cv=cv)
RMSE_lasso = sqrt(mean_squared_error(y,y_pred))
R2_lasso = r2_score(y,y_pred)
print("RMSE: " + str(round(sqrt(mean_squared_error(y,y_pred)),3)))
print("R_squared: " + str(round(r2_score(y,y_pred),3)))
```

5.1.3 Métodos de Predicción

```
import pandas as pd
from math import sqrt
import seaborn as sns
from pandas import DataFrame
datos = pd.read_excel(r'C:\Users\Eugenio\Desktop\Carrera\Cuarto\Segundo cuatri\TFG\Datos_2021.xlsx')
datos
datos.drop('Fecha', axis=1, inplace=True)
datos
%% split de datos para la primera combinacion de variables
X = datos.drop(columns='Precio')[['Demanda', 'Carbón', 'Ciclo Combinado', 'Cogeneración y Resto', 'Eólica', 'Intercambios Internacionales']]
y = datos['Precio']
```

```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size = 0.3)
```

5.1.3.1 Regresión Lineal

```
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error, mean_squared_error
lr = linear_model.LinearRegression()
lr.fit(X_train, y_train)
lr.get_params(deep= True)
y_pred = lr.predict(X_test)
print("RMSE: " + str(round(sqrt(mean_squared_error(y_test,y_pred)),2)))
print("R_squared: " + str(round(r2_score(y_test,y_pred),4)))
print("MAE: " + str(round(mean_absolute_error(y_test,y_pred),4)))
print("MSE: " + str(round(mean_squared_error(y_test,y_pred),4)))
```

5.1.3.2 Árboles de Decisión

```
from sklearn.tree import DecisionTreeRegressor
DtReg = DecisionTreeRegressor(random_state = 0)


DtReg.fit(X_train, y_train)
y_pred = DtReg.predict(X_test)
print("RMSE: " + str(round(sqrt(mean_squared_error(y_test,y_pred)),2)))
print("R_squared: " + str(round(r2_score(y_test,y_pred),4)))
print("MAE: " + str(round(mean_absolute_error(y_test,y_pred),4)))
print("MSE: " + str(round(mean_squared_error(y_test,y_pred),4)))
```

5.1.3.3 K Vecinos Próximos

```
from sklearn.neighbors import NearestNeighbors, KNeighborsRegressor
from sklearn.model_selection import cross_val_predict
import matplotlib.pyplot as plt
error = []
for k in range(1,51):
    knn = KNeighborsRegressor(n_neighbors = k)
    y_pred = cross_val_predict(knn, X, y, cv=5)
    error.append(mean_squared_error(y, y_pred))
plt.plot(range(1,51),error)
knn = KNeighborsRegressor(n_neighbors = 5)
y_pred = cross_val_predict(knn, X_test, y_test, cv=5)
print("RMSE: " + str(round(sqrt(mean_squared_error(y_test,y_pred)),2)))
print("R_squared: " + str(round(r2_score(y_test,y_pred),4)))
print("MAE: " + str(round(mean_absolute_error(y_test,y_pred),4)))
```

```
print("MSE: " + str(round(mean_squared_error(y_test,y_pred),4)))
```

5.1.3.4 Máquinas de Soporte Vectorial

```
import sklearn.svm as svm
svmt = svm.SVR(kernel = 'rbf')
svmt.fit(X_train, y_train)
y_pred = svmt.predict(X_test)
print("RMSE: " + str(round(sqrt(mean_squared_error(y_test,y_pred)),2)))
print("R_squared: " + str(round(r2_score(y_test,y_pred),4)))
print("MAE: " + str(round(mean_absolute_error(y_test,y_pred),4)))
print("MSE: " + str(round(mean_squared_error(y_test,y_pred),4)))
```

5.1.3.5 Redes Neuronales

```
import numpy as np
import pathlib
import keras
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
X = datos[['Demanda', 'Carbón', 'Ciclo Combinado', 'Cogeneración y Resto', 'Eólica', 'Intercambios Internacionales', 'Precio']]
train_dataset = X.sample(frac = 0.8, random_state = 0)
test_dataset = X.drop(train_dataset.index)
train_labels = train_dataset.pop('Precio')
test_labels = test_dataset.pop('Precio')
X_train_labels = X_train.pop
X_test_labels = X_test.pop
def build_model():
  model = keras.Sequential([
    layers.Dense(100, activation = tf.nn.relu, input_shape = [len(train_dataset.keys())]),
    layers.Dense(100, activation = tf.nn.relu),
    layers.Dense(1)
  ])

  optimizer = tf.keras.optimizers.RMSprop(0.001)

  model.compile(loss = 'mse',
                optimizer = optimizer,
                metrics = ['mae', 'mse'])
```

```
    return model
model = build_model()
model.summary()
example_batch = train_dataset[:10]
example_result = model.predict(example_batch)
example_result
class PrintDot(keras.callbacks.Callback):
    def on_epoch_end(self, epoch, logs):
        if epoch % 100 == 0: print('')
        print('.', end='')

EPOCHS = 1000

history = model.fit(
    train_dataset,train_labels,
    epochs = EPOCHS,validation_split = 0.2, verbose = 0,
    callbacks = [PrintDot()])
hist = pd.DataFrame(history.history)
hist['epoch'] = history.epoch
hist.tail()
def plot_history(history):
    hist = pd.DataFrame(history.history)
    hist['epoch'] = history.epoch

    plt.figure()
    plt.xlabel('Epoch')
    plt.ylabel('Error Medio Absoluto [Precio]')
    plt.plot(hist['epoch'], hist['mae'],
             label = 'Train Error')
    plt.plot(hist['epoch'], hist['val_mae'],
             label = 'Val Error')
    plt.legend()
    plt.ylim([0,75])

    plt.figure()
    plt.xlabel('Epoch')
    plt.ylabel('Error Cuadrático Medio [$Precio^2$]')
    plt.plot(hist['epoch'], hist['mse'],
```

```
        label = 'Train Error')
    plt.plot(hist['epoch'], hist['val_mse'],
        label = 'Val Error')
    plt.legend()
    plt.ylim([0,250])

plot_history(history)
loss, mae, mse = model.evaluate(test_dataset, test_labels, verbose = 0)
print("Error Absoluto Medio del Testing Set: {:5.2} Euros".format(mae))
print("Error Cuadrático Medio del Testing Set: {:5.2}".format(mse))
```

REFERENCIAS

7 Hidden Layers. (s.f.). *7 Hidden Layers*. Obtenido de 7 Hidden Layers: https://7-hiddenlayers.com/deep-learning-3/

APD, R. (4 de abril de 2019). *APD*. Obtenido de APD: https://www.apd.es/algoritmos-del-machine-learning/

Autor. (2012). Este es el ejemplo de una cita. *Tesis Doctoral, 2*(13).

Autor, O. (2001). Otra cita distinta. *revista*, pág. 12.

B12, E. d. (10 de Octubre de 2019). *B12 Tech4Business*. Obtenido de B12 Tech4Business: https://agenciab12.com/noticia/que-es-ciencia-de-datos

Barrán, A. T. (24 de abril de 2019). *Albertotb.* Obtenido de Albertotb: http://albertotb.com/curso-ml-R/Rmd/07-svm/07-svm.html#1

Bookdown. (s.f.). *Bookdown*. Obtenido de Bookdown: https://bookdown.org/content/2031/arboles-de-decision-parte-i.html

CARDONA M., C. A., & VELÁSQUEZ H., J. D. (julio de 2006). *redalyc.* Obtenido de redalyc: https://www.redalyc.org/pdf/496/49614914.pdf

Ciucci, M. (11 de 2020). *Fichas temáticas sobre la Unión Europea*. Obtenido de Fichas temáticas sobre la Unión Europea: https://www.europarl.europa.eu/factsheets/es/sheet/45/el-mercado-interior-de-la-energi

CNMC. (Mayo de 2021). *CNMC.* Obtenido de CNMC: https://www.cnmc.es/la-nueva-factura-de-la-luz

DataEvo. (2 de julio de 2018). *DataEvo*. Obtenido de DataEvo: https://www.dataevo.com.ar/post/diagrama-de-venn

Economipedia. (2 de octubre de 2017). Obtenido de Economipedia: https://economipedia.com/definiciones/r-cuadrado-coeficiente-determinacion.html

Endesa. (31 de 12 de 2019). *Endesa*. Obtenido de Endesa: https://www.endesa.com/es/sobre-endesa/quienes-somos/sociedades

Germán Morales España, J. M. (2008). Estrategia de regresión basada en el método de los k vecinos más cercanos para la estimación de la distancia de falla en sistemas radiales. *Revista de la Facultad de Ingeniería de la Universidad de Antioquia*, 9.

Gong, S. (30 de agosto de 2019). *Medium.* Obtenido de Medium: https://gongster.medium.com/how-does-a-neural-network-work-intuitively-in-code-f51f7b2c1e3f

González, A. (s.f.). *Cleverdata*. Obtenido de Cleverdata: https://cleverdata.io/que-es-machine-learning-big-data/

IBM. (2020). *IBM*. Obtenido de IBM: https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=networks-neural-model

IEBS. (2019). *IEBS*. Obtenido de IEBS.

Malato, G. (5 de mayo de 2021). *Towards Data Science*. Obtenido de Towards Data Science: https://towardsdatascience.com/feature-selection-in-machine-learning-using-lasso-regression-7809c7c2771a

Máxima Formación. (s.f.). *Máxima Formación*. Obtenido de Máxima Formación: https://www.maximaformacion.es/blog-dat/analisis-de-correlacion-en-r/

MBIT School. (30 de noviembre de 2020). *MBIT School*. Obtenido de MBIT School: https://www.mbitschool.com/que-es-data-science/

Miralles, Á. R. (2017). *MÉTODOS DE PREDICCIÓN APLICADOS AL PRECIO ELÉCTRICO.* Madrid: Escuela Politécnica Superior. Universidad Autónoma de Madrid.

Niño, M. (14 de julio de 2015). *El blog de Mikel Niño*. Obtenido de El blog de Mikel Niño: http://www.mikelnino.com/2015/10/big-data-origen-tecnologias-principales.html

Ochoa, M. O. (30 de abril de 2020). *Rpubs*. Obtenido de Rpubs: https://rpubs.com/movo/607465

OMIE. (2021). *OMIE.* Obtenido de OMIE: https://www.omie.es/es/sobre-nosotros

OMIE. (2021). *OMIE*. Obtenido de OMIE: https://www.omie.es/es/mercado-de-electricidad

Pandas. (s.f.). *Pandas*. Obtenido de Pandas: https://pandas.pydata.org/docs/index.html

Programo Ergo Sum. (s.f.). *Programo Ergo Sum*. Obtenido de Programo Ergo Sum: https://www.programoergosum.com/cursos-online/raspberry-pi/244-iniciacion-a-python-en-raspberry-pi/que-es-python

Raschka, S. (2014). *GitHub.* Obtenido de GitHub: http://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector/

Salazar, C. (13 de febrero de 2018). *Rpubs*. Obtenido de Rpubs: https://rpubs.com/camilamila/correlaciones