

**UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR**



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**Métodos de aprendizaje automático para la
predicción de demanda eléctrica**

Autor: Carlos Aliaga Samper

Tutor: José Ramón Dorronsoro Ibero

mayo 2024

Carlos Aliaga Samper
Métodos de aprendizaje automático para la predicción de demanda eléctrica

Carlos Aliaga Samper

IMPRESO EN ESPAÑA – PRINTED IN SPAIN

RESUMEN

El mundo en el que vivimos en la actualidad necesita en todo momento del flujo de suministro eléctrico. Desde las acciones más triviales de nuestro día a día como encender la luz hasta los procesos más complejos que pueden tener lugar en distintas fábricas o laboratorios. El hecho de contar con energía eléctrica es algo que se tiene tan interiorizado que ni se plantea la posibilidad de que, en algún momento, ésta deje de estar ahí.

Sin embargo, para que toda la población disponga de electricidad, hay una gran organización detrás que asegura que así sea. El sistema eléctrico es un sistema complejo principalmente por una razón: la energía eléctrica no se puede almacenar en grandes cantidades. Por ello, se debe asegurar en todo momento que la generación y la demanda vayan de la mano, sean lo más parecidas posible.

Con esta idea en mente, se plantea el mundo del sector energético como un gran lugar donde aplicar los conocimientos sobre aprendizaje automático para contribuir de manera positiva en el día a día de la sociedad. En ese sentido, en este trabajo se proponen algoritmos de aprendizaje automático como solución al problema de la estimación de la demanda de energía eléctrica. Más concretamente, se estudiarán distintos modelos lineales, redes neuronales y SVM, la adición de datos de temperatura y distintos procesos de normalización de los datos.

PALABRAS CLAVE

Sistema eléctrico, Demanda eléctrica, Predicción, Normalización, Aprendizaje automático, Ridge Regression, Multilayer Perceptron, SVR, Test de Wilcoxon

ABSTRACT

The world we live in nowadays constantly relies on the flow of electrical supply. From the most trivial actions of our daily lives like turning on the lights to the most complex processes that can take place in different factories or laboratories. The availability of electrical energy is so ingrained that the possibility of it not being there at some point cannot even be considered.

However, for the entire population to have access to electricity, there is a large organization behind ensuring that this is the case. The electrical system is a complex system, and it is primarily for one reason: electrical energy cannot be stored in large quantities. Therefore, it must be ensured at all times that generation and demand go hand in hand, as closely as possible.

With this idea in mind, the world of the energy sector is presented as a great place to apply machine learning methods to contribute positively to society's daily life. In this regard, this work proposes machine learning algorithms as a solution to the problem of estimating electricity demand. More specifically, different linear models, neural networks, and *SVM* will be studied, along with the addition of temperature data and various data normalization processes.

KEYWORDS

Electrical system, Electrical demand, Prediction, Normalization, Machine learning, Ridge Regression, Multilayer Perceptron, SVR, Wilcoxon test

ÍNDICE

1	Introducción	1
1.1	Objetivos	1
1.2	Estructura del trabajo	1
2	El sistema eléctrico	3
2.1	El concepto de sistema eléctrico	3
2.2	Normalización	9
3	Modelos principales	17
3.1	Métodos lineales	17
3.2	Redes neuronales	21
3.3	Support Vector Machines	25
4	Experimentos	29
4.1	Descripción del entorno	29
4.2	Obtención y procesado de los datos	30
4.3	Análisis de los datos normalizados	31
4.4	Experimentos	32
5	Conclusiones y trabajo futuro	39
5.1	Conclusiones	39
5.2	Trabajo futuro	40
	Bibliografía	42
	Apéndices	43
A	Resultados del procesado de los datos	45
B	Hiperparametrizaciones	49
C	Tests estadísticos	65

INTRODUCCIÓN

1.1. Objetivos

Casi todo lo que encontramos a nuestro alrededor necesita de electricidad para funcionar: el frigorífico donde guardamos los alimentos, la televisión delante de la cual pasamos el rato, la bombilla que nos ilumina al leer, el ordenador con el que trabajamos... Y así podríamos enumerar una enorme sucesión de cosas que tenemos tan asimiladas que están ahí siempre a nuestra disposición que entraríamos en un gran caos si, de repente, dejara de llegar a nosotros la electricidad.

Pero, ¿cómo llega a nosotros?, ¿qué hace posible todos esos gestos cotidianos? En un principio, podríamos pensar que alguien se encarga de producirla, se almacena y, cuando es necesaria, la obtenemos de donde la hayamos dejado. Si bien esta es la cadena de cualquier otro bien físico que podamos adquirir, con la electricidad falla un eslabón de la cadena: no se puede almacenar en grandes cantidades. Por ello, se ha de encontrar un equilibrio, en tiempo real y las 24 horas del día los 365 días del año, entre la generación y la demanda de energía eléctrica. Para obtenerlo, no hay otra opción que realizar una previsión de la demanda lo más exacta posible. Este hecho convierte al sistema eléctrico en un sistema complejo y es la base de este trabajo.

El objetivo principal del TFG es desarrollar métodos para la predicción de la demanda de energía eléctrica. Para ello, se estudiarán distintos modelos de aprendizaje automático, la adición de datos sobre temperatura y diversos procesos de normalización de los datos para mejorar los resultados. Para la consecución de dicho objetivo se ha desarrollado código *Python*, que puede ser consultado y descargado desde el siguiente repositorio de *GitHub*: <https://github.com/CarlosAliagaSamper/TFG-Informatica>.

1.2. Estructura del trabajo

Este trabajo consta de cinco capítulos. El primero de ellos es esta sección, donde se realiza una introducción al mismo, se exponen sus objetivos y se describe su estructura.

En el capítulo 2 se explicará el concepto de sistema eléctrico, explicando los componentes y agentes que lo conforman y destacando el papel fundamental que juega la estimación de la demanda para encontrar el necesario equilibrio entre generación y demanda. También se indagará en las características específicas del sistema eléctrico español. Adicionalmente, se expondrán las características de los datos de demanda de energía eléctrica y posibles procesos para su normalización basados en ellas.

Posteriormente, en el capítulo 3 se propondrán modelos de *machine learning* como herramientas a usar para la estimación de la demanda energética. En ese sentido, se presentarán los modelos lineales, estudiando en mayor profundidad los modelos basados en *Ridge Regression*, las redes neuronales (*Multilayer Perceptrons*) y las *Support Vector Machines* adaptadas a problemas de regresión (SVR).

Después, en el capítulo 4 se explicarán los experimentos prácticos realizados sobre datos reales de demanda de energía eléctrica en España. En ese sentido, se explicará desde la elección del entorno de programación sobre el que se trabajó hasta la discusión sobre los resultados propiamente dichos, pasando por la obtención y procesado de los datos, y el análisis de los resultados de los procesos de normalización previos a las estimaciones.

Finalmente, en el capítulo 5 se recapitulará todo el trabajo realizado. En ese sentido, se expondrán las conclusiones finales del trabajo y se propondrán posibles campos en los que este trabajo podría ser ampliado en un futuro.

Adicionalmente, se adjuntan 3 anexos que complementan los desarrollos realizados en el cuerpo del trabajo. Por una parte, en el anexo A se adjuntan figuras sobre los resultados del tratamiento de los datos. Se trata de gráficas e histogramas que complementan el análisis del procesado de dichos datos. Por su parte, en el anexo B se adjuntan los resultados de las hiperparametrizaciones realizadas sobre los modelos empleados para la estimación de la demanda de energía eléctrica. Para terminar, en el anexo C se adjuntan los tests estadísticos realizados sobre los resultados de los experimentos prácticos del trabajo. Dichos tests intentan comprobar si existe suficiente evidencia estadística para poder afirmar que un modelo obtiene mejores resultados que otro.

EL SISTEMA ELÉCTRICO

Un sistema eléctrico es un conjunto complejo de elementos que han de colaborar entre sí para que la energía eléctrica llegue con normalidad a los consumidores finales. Engloba desde los procesos de generación y transporte hasta la distribución final. Para garantizar dicha coordinación, existe también el rol de operador del sistema, que es la organización encargada de ello.

En este capítulo, en la sección 2.1, se desarrollará el concepto de sistema eléctrico y cuáles son los elementos concretos que lo conforman, junto con las características más concretas del sistema eléctrico español, orientándose a su operación y a la previsión de la demanda. Dicho desarrollo está basado en lo expuesto en [1]. También se presentará, a su vez, el papel clave que juega el mercado eléctrico en estos procesos.

Por otra parte, en la sección 2.2, se presentarán las características especiales de los datos de demanda energética, cuya principal particularidad es su heterogeneidad. Basadas en dichas características, se propondrán también algunas ideas para su normalización, muy necesarias para la posterior precisión de su previsión debido a dicha heterogeneidad que poseen sus datos.

2.1. El concepto de sistema eléctrico

Un sistema eléctrico se puede definir como el conjunto de elementos que operan de manera coordinada en un área específica con el fin de satisfacer la demanda de energía eléctrica. En este sentido, sus elementos más habituales son los siguientes:

- Los centros o plantas de generación, que producen la electricidad y elevan su tensión para el transporte.
- Las líneas de transporte de alta tensión.
- Las estaciones transformadoras, que reducen la tensión.
- Las líneas de distribución de media y baja tensión, que llevan la electricidad hasta los usuarios finales.
- Las instalaciones de los clientes de energía eléctrica.
- Los centros de control de las empresas generadoras, distribuidoras y comercializadoras.
- Un centro de control eléctrico nacional, que gestiona, coordina y opera todo el sistema eléctrico. Es el encargado de conseguir el equilibrio entre la generación y demanda de energía eléctrica.

Además de estos elementos, un sistema eléctrico necesita de la acción de diversos actores:

- Los generadores: son los encargados de la producción de la energía eléctrica.
- El transportista: es la empresa encargada de transportar dicha electricidad desde su lugar de producción a los centros de consumo de los distribuidores.
- El operador del sistema: es la empresa encargada de conseguir el equilibrio entre la generación y la demanda de energía eléctrica.
- Los distribuidores: son las empresas encargadas de llevar la electricidad al consumidor final.
- Los comercializadores: son las empresas que compran la energía eléctrica en el mercado eléctrico para venderla a los consumidores.
- Los consumidores: son los destinatarios de todo el proceso, los que consumen la electricidad.

A continuación, en las siguientes secciones, se presentan las características concretas del sistema eléctrico español.

2.1.1. El sistema eléctrico español

Una primera distinción especial que se debe remarcar del sistema eléctrico español es que está compuesto, a su vez, por varios sistemas eléctricos: el peninsular y los no peninsulares (los correspondientes a las Islas Baleares, a las Islas Canarias, a Ceuta y a Melilla).

Con el objetivo de no sobrecomplicarlo, este trabajo se restringirá al sistema peninsular. Por este motivo, cuando a partir de este momento se mencione el sistema eléctrico español se estará refiriendo solamente al peninsular, a no ser que se especifique lo contrario.

También se debe mencionar qué empresas u organizaciones juegan cada papel de los descritos en la sección anterior en el caso concreto del sistema eléctrico español. En general, en España, hay grandes empresas eléctricas, como Endesa o Iberdrola entre otras, que se dedican tanto a la generación como a la comercialización de la energía eléctrica. Además de ellas, hay otras empresas que están dedicadas solamente a una de las dos actividades mencionadas anteriormente. En este sentido, cabe destacar la existencia de empresas que se especializan y dedican de forma única a la generación de energía mediante fuentes renovables.

Por otra parte, las empresas distribuidoras en España son únicas en cada territorio. Dependiendo de en qué parte de la geografía peninsular nos encontremos, la distribuidora de energía eléctrica puede variar, pero en ese lugar es única. Hay muchas de ellas, pero las que trabajan la mayor parte de la superficie española son Endesa, Iberdrola, E-Redes Distribución (Grupo EDP), Unión Fenosa y Viesgo Distribución (EON).

El otro gran actor del sistema eléctrico español es Red Eléctrica de España que, como se verá en la próxima sección donde se la presentará, es la empresa encargada dentro del sistema eléctrico español, en su totalidad, de las líneas de transporte de alta tensión, gestionar el centro de control eléctrico nacional y operar el sistema eléctrico.

2.1.2. Redeia - Red Eléctrica de España

Redeia - Red Eléctrica de España, REE de aquí en adelante, es la empresa encargada del transporte de energía eléctrica en España y la operación del sistema eléctrico español, tanto del sistema peninsular como de los sistemas no peninsulares.

Nació en 1985 en aplicación de la Ley 49/1984, de 26 de diciembre [2]. Esta ley establecía que la explotación unificada del sistema eléctrico nacional a través de una red de alta tensión es un servicio público de titularidad estatal, encargando a una nueva sociedad estatal estas tareas. Red Eléctrica de España es dicha sociedad estatal. Posteriormente, con nuevas leyes, REE se adaptó a la normativa europea y se confirmó como transportista y operador único del sistema eléctrico español.

Desde su creación, el principal objetivo de REE fue claro: garantizar en todo momento la seguridad y continuidad del suministro eléctrico de nuestro país y desarrollar una red de transporte fiable que contribuya al progreso de la sociedad [3].

En definitiva, REE tiene dos principales funciones. La primera de ellas, en su condición de gestor de la red de transporte, es transportar la energía eléctrica en alta tensión desde los centros de generación hasta las zonas de consumo. Por ello, se encarga de gestionar las infraestructuras eléctricas que componen la red de transporte propia y del desarrollo, mantenimiento y ampliación de dicha red con criterios homogéneos y eficientes. Asimismo, es responsable de gestionar el tránsito de energía de entre la Península y sistemas exteriores vecinos y de garantizar el acceso de terceros a la red en condiciones de igualdad.

La segunda, en su condición de operador del sistema eléctrico, es garantizar en todo momento la seguridad y continuidad del suministro, a través de sus centros de control, tanto en el sistema peninsular como en los no peninsulares. En relación a esta segunda función, se presenta a continuación la operación del sistema eléctrico español [4].

2.1.3. La operación del sistema eléctrico español

La operación del sistema eléctrico comprende las actividades necesarias para garantizar la seguridad y continuidad del suministro eléctrico, así como la correcta coordinación entre el sistema de producción y la red de transporte, asegurando que la energía producida por los generadores sea transportada hasta las redes de distribución con las condiciones de calidad exigibles en aplicación de la normativa vigente [5].

Como se ha visto antes, una de las características de la energía eléctrica es que no puede almacenarse para su posterior consumo. Esto supone que, para el correcto funcionamiento del sistema eléctrico, la producción de las centrales de generación debe igualarse, o acercarse al máximo posible, al consumo de forma instantánea.

Aquí es donde entra el papel de Red Eléctrica de España como operador del sistema eléctrico. REE realiza las previsiones de la demanda de energía y gestiona en tiempo real las instalaciones de generación y transporte, logrando que la producción programada en las centrales eléctricas coincida en cada momento con la demanda de los consumidores. En caso de diferir, envía las órdenes oportunas a las centrales para que ajusten sus producciones a los cambios en el consumo previsto.

En su labor como operador del sistema eléctrico, REE realiza diversas tareas. Dichas tareas se pueden separar en tres clases, dependiendo de la finalidad que tengan o a qué estén orientadas: para la previsión de demanda a largo plazo, a medio plazo o a corto plazo. En este trabajo se indagará sobre la previsión de demanda a corto plazo.

En su función de operador del sistema eléctrico español, el objetivo principal de Red Eléctrica de España es cubrir la demanda eléctrica que los clientes hagan en su totalidad. Para lograrlo, busca la mayor coordinación posible entre todos los actores que conforman el sistema eléctrico. También, en caso de cualquier fallo en cualquier eslabón de la cadena, REE está preparada para sobrellevarlo.

Gracias a la colaboración de todos los actores del sistema eléctrico, el operador del mismo (REE) es capaz de captar los datos precisos para conocer en tiempo real el consumo de energía eléctrica. A su vez, Red Eléctrica de España hace también públicos estos datos de forma gráfica a través de su página web. Para poder ofrecer esta coordinación al resto de los actores, REE lleva a cabo las siguientes tres tareas, que se podrían clasificar en la órbita de la previsión de demanda a largo plazo:

- La previsión del consumo eléctrico futuro, que consiste en calcular las necesidades de generación y de red a largo plazo. Este cálculo no es fijo, puede ir variando con el tiempo con el fin de ajustarse más a la realidad.
- La determinación de la potencia de generación a instalar, es decir, cuál será el nivel de producción necesario para satisfacer las necesidades de la sociedad.
- La planificación de la red de transporte, donde se planifica cómo se va a llevar esa energía desde los centros de generación hasta los usuarios finales.

En verdad, la primera tarea de la lista rige las otras dos. En función de la previsión del consumo eléctrico que se haga, se planificará la infraestructura que puede ser necesaria para su producción y posterior transporte.

2.1.4. La previsión de la demanda

Para la tan deseada y necesaria previsión de la demanda, Red Eléctrica de España cuenta con distintas series de datos históricos como, entre otros:

- La demanda real de energía eléctrica que ha realizado la población en el pasado.
- Las previsiones de evolución de la economía.
- Distintas previsiones meteorológicas, como la temperatura o la nubosidad.
- Datos sobre régimen y horarios laborables en los distintos territorios de nuestro país.

Con esta información, haciendo uso de distintas aplicaciones informáticas y algoritmos matemáticos, realizan una estimación de la demanda.

Primero se realizan las previsiones de demanda a largo plazo, que son las que permiten estimar las necesidades de generación y red de transporte. En estas previsiones tienen una gran influencia las previsiones de evolución de la economía y los nuevos usos eléctricos.

Del mismo modo que se realizan las programaciones anuales, mensuales y semanales, el operador del sistema eléctrico debe prever cada día qué demanda va a alcanzarse para cada hora del día siguiente.

Partiendo de estas primeras previsiones a largo plazo, se realizan unas previsiones a medio y corto plazo (cada semana, cada día, cada hora, ...) para ajustarse más a la realidad. En este momento es en el que tienen un papel más principal la previsión de la temperatura o la laboralidad. Es en esta parte de la previsión sobre la que se realizarán todos los posteriores experimentos prácticos.

La diferencia con la previsión a largo plazo es que ahora no se fija, en un principio, el programa horario que han de producir los generadores. En este momento entra en acción un nuevo actor: el mercado de electricidad.

2.1.5. El mercado eléctrico

En el sistema eléctrico español, la cobertura de la demanda se realiza a través del llamado Mercado Ibérico de la Electricidad, MIBEL, en el que concurren:

- Los generadores, que presentan ofertas de venta de la energía eléctrica que tienen disponible.
- Los comercializadores, que presentan sus ofertas de compra de la energía eléctrica que ellos consideren que necesitan.
- Los sujetos usuarios de las interconexiones internacionales, que compran o venden energía estableciendo intercambios internacionales.
- Algunos consumidores directos, que realizan la adquisición de energía en el propio mercado.

Este mercado, diario, es gestionado por el Operador del Mercado Ibérico de Energía, OMIE, que define los programas de producción y consumo para cada hora del día siguiente. Éste se rige dando prioridad a los grupos de generadores que mejores precios hayan ofertado.

Una vez que OMIE publica el resultado del programa diario base de funcionamiento del día siguiente, el operador del sistema, REE, lleva a cabo un análisis con ayuda de aplicaciones informáticas de simulación para determinar si el sistema eléctrico es seguro o si se da una situación de riesgo en la que podrían producirse apagones.

Si el programa no es viable, el operador del sistema eléctrico lo modificará, mediante el proceso conocido como “gestión de las restricciones técnicas del sistema”, que tiene por objetivo garantizar que el programa económico derivado de la libre contratación en los mercados, el presentado por OMIE, sea técnicamente factible.

Para poder llevar a cabo estos ajustes, los generadores fijan en un principio su producción a un

punto medio, desde el cual pueden oscilar su producción entre dos puntos, la potencia nominal y el mínimo técnico, para que el equilibrio entre demanda y generación se mantenga. A esta reserva se le llama regulación secundaria, y es aquella que los generadores pueden aportar entre 2 y 15 minutos después de un incidente. Existe también una regulación primaria, la proporcionada por los generadores gracias a su propia inercia. Tras la programación de la reserva secundaria, se completa el programa del día siguiente. Éste es el plan de producción que se ejecutará el día siguiente.

Cada día, desde el Centro de Control Eléctrico, CECOEL, se vigila, controla, coordina y teleopera todo el sistema eléctrico durante las 24 horas. Desde el CECOEL también se analiza en cada momento la situación, determinando si es necesario tomar medidas de operación adicionales para asegurar la seguridad del sistema eléctrico.

Además del mercado diario, cada día se realizan seis sesiones del mercado intradiario, también gestionadas por el operador del mercado, a las que acuden los generadores a retirar su energía programada o vender más energía y los comercializadores a deshacerse de parte de la energía que habían comprado o a comprar más (dependiendo de la situación).

Tras cada sesión del mercado intradiario, el CECOEL vuelve a analizar el nuevo escenario para verificar que se sigue asegurando la seguridad del sistema eléctrico. Cuando se prevé un desequilibrio demasiado grande, alrededor de los 300 MWh, entre la generación y la demanda, el propio operador del sistema eléctrico puede convocar un nuevo tipo de mercado: el mercado de gestión de desvíos, al que tan solo acuden los generadores para bien aumentar su producción o bien disminuirla para ajustarse a la demanda.

Para poder llevar a cabo una operación segura y económica del sistema eléctrico español, Red Eléctrica de España ha desarrollado un sistema de información, que recibe el nombre de Sistema de Información del Sistema, e-sios. El e-sios permite:

- Comunicarse con el operador del mercado, para intercambiar los resultados de los distintos mercados.
- Comunicarse con los sujetos del mercado, quienes acuden con ofertas de compra o venta de energía y se les notifica la aceptación o rechazo de sus ofertas.
- Comunicarse con los técnicos del propio operador del sistema.
- Comunicarse con los operadores de los sistemas eléctricos vecinos.
- Publicar los resultados de los distintos mercados y programaciones.
- Almacenar en su base de datos histórica toda la información.

De hecho, es de este sistema e-sios desde el que se obtendrán posteriormente los datos para el estudio práctico.

2.2. Normalización

Como se ha visto anteriormente, para satisfacer el objetivo final de encontrar el equilibrio entre generación y demanda de energía eléctrica, la clave reside en predecir de la forma más precisa posible la demanda que va a ejercer la población en el futuro próximo. Sin embargo, como se puede apreciar en las figuras 2.1 y 2.2, los datos históricos que se manejan son muy heterogéneos.

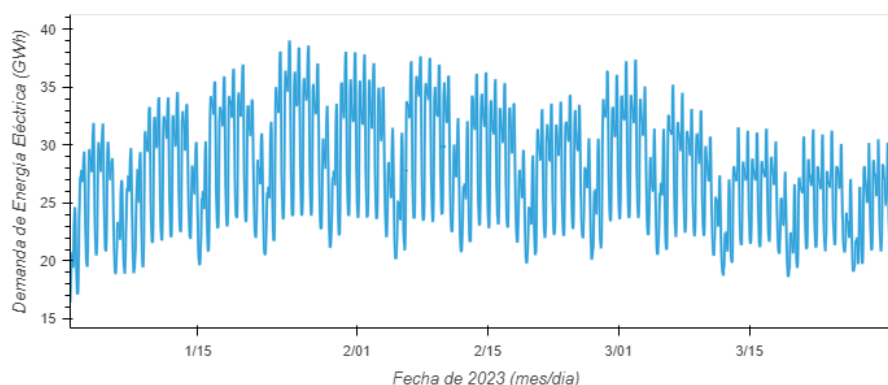


Figura 2.1: Demanda de energía eléctrica entre enero y marzo de 2023 en España

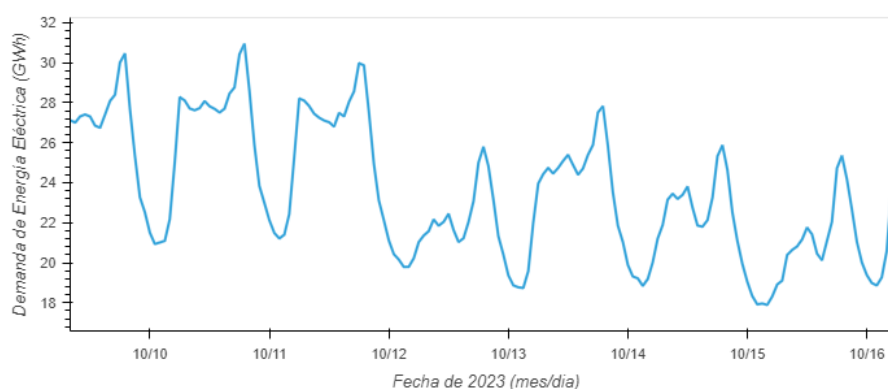


Figura 2.2: Demanda de energía eléctrica la semana del 9 al 15 de octubre de 2023 en España

Se puede apreciar cómo distintas variables, como el día de la semana o la laboralidad, afectan directamente a la demanda de energía eléctrica. Pero no solamente eso, estas variables no afectan a la demanda de la misma manera durante todas las horas del día, como puede apreciarse en la figura 2.3.

Con el propósito de homogeneizar los datos surge la normalización, es decir, el intento de tratar los datos para estandarizarlos y predecir datos futuros, que pueden volver a transformarse para seguir la ley de probabilidad original que los genera, con una mayor precisión.

Basándonos en las características concretas de los datos de demanda de energía eléctrica presentados anteriormente, se exponen a continuación dos ideas distintas para su normalización. Para expresarlas formalmente, se definen a continuación los siguientes conjuntos:

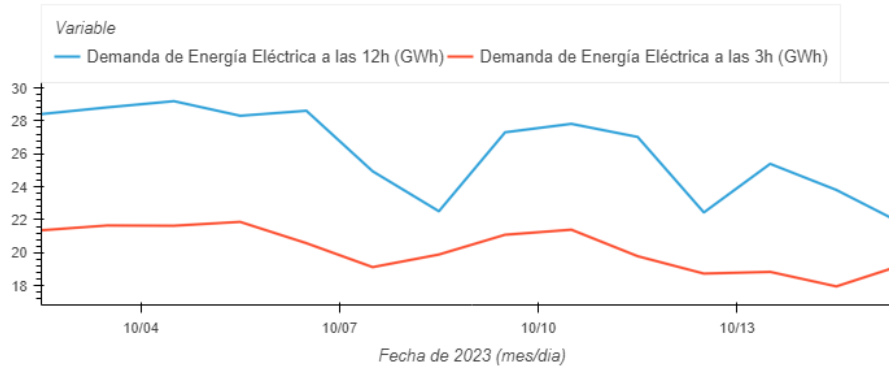


Figura 2.3: Demanda de energía eléctrica entre los días 2 y 16 de octubre de 2023 en España a las 3h y a las 12h

- D : El conjunto de los días sobre los que se van a realizar las mediciones, es decir, en este trabajo, $D = \{2021/01/01, \dots, 2023/12/31\}$, donde los días están en formato año/mes/día.
- H : El conjunto de las horas del día, es decir, $H = \{00h, \dots, 23h\}$.

Se ha elegido este rango de días ya que es el más amplio conteniendo años completos y con datos significativos, ya que en 2020 con la pandemia del COVID-19 esos datos no reflejarían la realidad de una situación normal.

Posteriormente, se dividirá el conjunto D en otros subconjuntos, distintos para cada método de normalización a aplicar. También, para facilitar la escritura de las fórmulas, se empleará la notación $V_{d,h}$ para expresar el valor de demanda energética el día d a la hora h . Dicho día y hora asociados están en el huso horario UTC.

2.2.1. Primera idea: aproximación a día laborable

La primera idea para normalizar los datos se basa en la definición de un día común. Con ese propósito, se definen primero los siguientes subconjuntos de D :

- Lab : El conjunto de los días de entre semana completamente laborables en España.
- Sab : El conjunto de los sábados completamente laborables en España.
- Fes : El conjunto de los días completamente festivos en España junto con los domingos.

Nótese que estos tres subconjuntos, Fes , Lab y Sab , no forman una partición del conjunto D . Aunque sí que son disjuntos, su unión no forma el conjunto total D . Por ejemplo, tomemos el día $2023/05/02 \in D$. Por una parte, el 2 de mayo de 2023 fue martes, luego $2023/05/02 \notin Sab$. Y por otra parte, el 2 de mayo es festivo en la Comunidad de Madrid, luego $2023/05/02 \notin Lab$, pero no en el resto de España, luego $2023/05/02 \notin Fes$.

Sin embargo, se puede asumir que en estos casos se cumple que el día d es, aproximadamente,

combinación lineal de una parte completamente laborable (l_d si es un día de entre semana ó s_d si es un sábado) y una parte completamente festiva f_d . En este sentido, si $\rho_d \in [0, 1]$ es el valor que ajusta el siguiente modelo, es decir, ρ_d es la fracción de población en España que trabaja el día d , podemos entender las siguientes “ecuaciones”, que expresan dicha combinación lineal:

$$d = \rho_d * l_d + (1 - \rho_d) * f_d, \quad (2.1)$$

cuando d es un día de entre semana o domingo, ó

$$d = \rho_d * s_d + (1 - \rho_d) * f_d, \quad (2.2)$$

cuando d es un sábado.

Nótese los casos particulares en los que $\rho_d = 0$ ó $\rho_d = 1$. El primero de ellos se refiere a que el día d es un día completamente festivo o domingo y, por tanto, como cabría esperar, tanto la ecuación (2.1) como la ecuación (2.2) quedarían:

$$d = f_d.$$

De forma similar, $\rho_d = 1$ indica que d es un día completamente laborable, y las ecuaciones (2.1) y (2.2) quedarían como sigue:

$$d = l_d,$$

cuando d es un día de entre semana, ó

$$d = s_d,$$

cuando d es un sábado; luego se pueden extender ambas ecuaciones al conjunto total de días D . Trasladando estas hipótesis sobre los días a la demanda, apoyándose en la idea de que la demanda de energía eléctrica tiende a ser similar los días de entre semana, pero diferente a sábados, domingos y festivos (siendo el caso de los sábados a su vez distinto del último caso), surge la siguiente idea.

Se elige como día estándar un día de entre semana completamente laborable, ya que es el más común de todos los tipos. La idea es, de cada día, obtener su componente de día de entre semana completamente laborable, o en el caso de los sábados, obtener su componente de sábado completamente laborable y transformarla en la esperada si fuera un día entre semana.

Para expresar formalmente la normalización, se definen las siguientes funciones auxiliares, que expresan el coeficiente de normalización por el que habría que multiplicar el valor real de la demanda energética dependiendo de si se trata de un sábado (C_{Sab}) u otro día de la semana (C_{Otro}). En ellas, $\bar{V}_{Lab,h}$, $\bar{V}_{Sab,h}$ y $\bar{V}_{Fes,h}$ representan el valor medio de demanda energética los días de entre semana completamente laborables, los sábados completamente laborables y los domingos y días completamente festivos respectivamente, a la hora h , y ρ_d es el valor explicado con anterioridad.

- $C_{Sab} : D \times H \rightarrow \mathbb{R}; \quad C_{Sab}(d, h) = \frac{\bar{V}_{Lab, h}}{\bar{V}_{Sab, h}} * \rho_d + \frac{\bar{V}_{Lab, h}}{\bar{V}_{Fes, h}} * (1 - \rho_d),$
- $C_{Otro} : D \times H \rightarrow \mathbb{R}; \quad C_{Otro}(d, h) = \rho_d + \frac{\bar{V}_{Lab, h}}{\bar{V}_{Fes, h}} * (1 - \rho_d).$

La primera observación que se ha de hacer es la similitud entre ambas funciones. La única diferencia se encuentra en el primero de los sumandos en el que, en la primera función ρ_d viene multiplicado por la proporción entre el valor medio de la demanda energética los días de entre semana completamente laborables y el valor medio de la demanda energética los sábados completamente laborables. El por qué viene dado por la lógica detrás de las funciones.

En ambos casos, el segundo de los sumandos obtiene la proporción del valor de la demanda energética explicada por su componente festiva y la transforma al valor esperado si fuera un día completamente laborable. Sin embargo, el primer sumando juega papeles diferentes en cada función.

En la función C_{Sab} se obtiene la proporción del valor de la demanda energética explicada por su componente de sábado completamente laborable y la transforma al valor esperado si fuera un día completamente laborable. Con esta última idea en la cabeza, se entiende cómo el primer sumando de la función C_{Otro} obtiene la proporción del valor de la demanda energética explicada por su componente de día de entre semana completamente laborable y la transforma al valor esperado si fuera un día completamente laborable, es decir, la multiplica por 1.

Nótese el ejemplo de la normalización trivial. Si se tiene el valor de la demanda de energía eléctrica de un día de entre semana completamente laborable a una cierta hora, cabría esperar que el coeficiente de normalización fuera 1. En este caso, como se verá en la siguiente función a definir, se utilizaría la función C_{Otro} para obtenerlo. Puede observarse cómo, al ser $\rho_d = 1$, como se esperaba, el valor del coeficiente de normalización es 1, es decir, el dato ya proviene de un día estándar.

Estas dos funciones permiten definir la siguiente función general para obtener el coeficiente de normalización:

$$C : D \times H \rightarrow \mathbb{R}; \quad C(d, h) = \begin{cases} C_{Sab}(d, h) & \text{si } d \text{ es sábado,} \\ C_{Otro}(d, h) & \text{en otro caso.} \end{cases}$$

Con esta última función, se puede expresar la normalización como sigue, recordando que $V_{d,h}$ representa el valor de la demanda energética el día d a la hora h :

$$Norm_1 : D \times H \rightarrow \mathbb{R}; \quad Norm_1(d, h) = V_{d,h} * C(d, h) = N_{d,h}. \quad (2.3)$$

Con este proceso, la posterior inversión de la normalización de los datos una vez predicha la demanda normalizada es muy simple: si $\hat{N}_{d,h}$ es la demanda normalizada predicha, entonces el valor de demanda de energía eléctrica predicha, $\hat{V}_{d,h}$, será el siguiente:

$$\hat{V}_{d,h} = \frac{\hat{N}_{d,h}}{C(d,h)},$$

resultado de despejar $V_{d,h}$ en la ecuación (2.3).

2.2.2. Segunda idea: semana previa

La segunda idea, y más simple, se basa en transformar los datos a la proporción con respecto a la demanda real una semana atrás a la misma hora. Esta idea surge a raíz de que, salvo en dos excepciones que se verán a continuación, cabe esperar que la demanda de un día concreto a una hora concreta sea muy parecida a la demanda del mismo día una semana atrás a la misma hora, como puede observarse en las figuras 2.4 y 2.5, correspondientes a la demanda de energía eléctrica en España en dos semanas consecutivas sin festividades nacionales. Por tanto, dicha proporción oscilará en valores cercanos a 1.

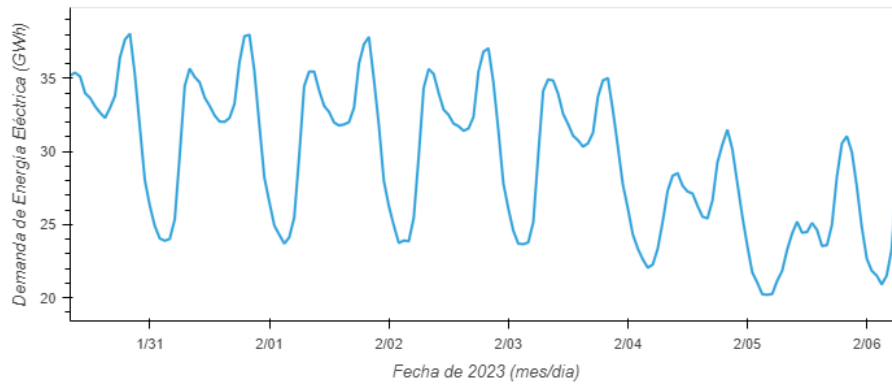


Figura 2.4: Demanda de energía eléctrica la semana del 30 de enero al 5 de febrero de 2023 en España

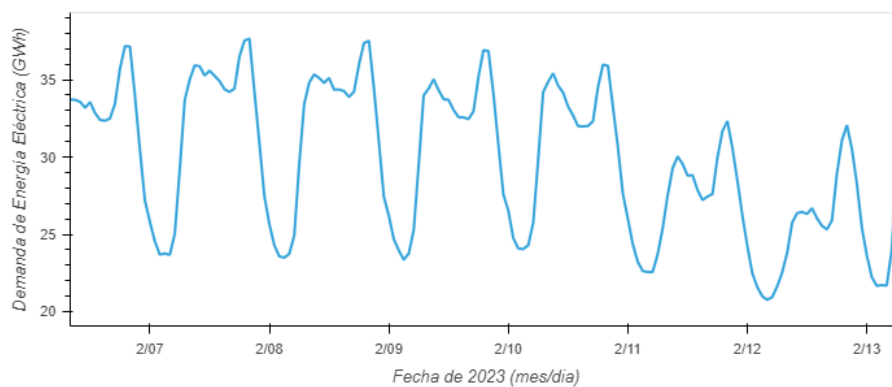


Figura 2.5: Demanda de energía eléctrica la semana del 6 al 12 de febrero de 2023 en España

Sin embargo, esta hipótesis es claramente falsa cuando uno de los días afectados en la normalización es festivo, es decir, cuando el día que se intenta normalizar, o el día con respecto al cual se toma la proporción, son festivos. Se tratan estos dos casos de forma diferente para solucionarlo.

Cuando nos encontremos en el caso en el que el día que se intenta normalizar es festivo la idea es buscar el festivo más cercano, y hallar la proporción con él. A efectos prácticos, la demanda de energía eléctrica los domingos es muy parecida a la demanda de un día festivo. Por ello, se normalizará los días festivos encontrando la proporción con respecto al domingo más cercano al día en cuestión.

Por otra parte, cuando nos encontremos en el caso en el que el día que se intenta normalizar es un día laborable, pero ese mismo día una semana atrás fue festivo nos retrasaremos otra semana hacia atrás, y así sucesivamente hasta encontrar una semana en la que ese día fue laborable. Para expresar formalmente estos pasos, se definen los siguientes subconjuntos de D :

- Fes : El conjunto de los días completamente festivos en España junto con los domingos.
- Lab : El resto de días, donde se incluyen tanto los días de entre semana laborables como los sábados laborables. Se incluyen esta vez en este conjunto también los días parcialmente festivos, es decir, los días que son festivos simplemente en alguna región de España.

En este caso, Fes y Lab sí que forman una partición del conjunto D . Este hecho será clave para la posterior definición de la función de normalización. A su vez, se define el conjunto Dom como el conjunto de los domingos, necesario para las próximas definiciones. Nótese que $Dom \subset Fes$.

Se definen ahora las siguientes funciones auxiliares, cuyo objetivo es encontrar el día respecto al cual se ha de realizar la proporción:

- $\Delta_{Lab} : Lab \rightarrow \mathbb{N}$; $\Delta_{Lab}(d) = \min_{n \in \mathbb{N}} \{7n : d - 7n \in Lab\}$,
- $\Delta_{Fes} : Fes \rightarrow \mathbb{N}$; $\Delta_{Fes}(d) = \min_{n \in \mathbb{N}} \{n : d - n \in Dom\}$,

En este sentido, se entiende la resta de un día d por un número natural n como el día resultado de ir n días atrás desde d . La función Δ_{Lab} devuelve el mínimo múltiplo positivo de 7, digamos $7m$, tal que al ir $7m$ días atrás desde d se encuentra un día laborable. Por otra parte, la función Δ_{Fes} devuelve el mínimo número natural n tal que al ir n días atrás desde d se encuentra un domingo.

Estas dos funciones permiten definir la siguiente función general, que está bien definida gracias al hecho de que Fes y Lab forman una partición de D . Esta nueva función general devuelve el día respecto al cual se ha de realizar la proporción para normalizar:

$$\Delta : D \rightarrow D; \quad \Delta(d) = \begin{cases} d - \Delta_{Lab}(d) & \text{si } d \in Lab, \\ d - \Delta_{Fes}(d) & \text{si } d \in Fes. \end{cases}$$

Con esta última función, se puede expresar la normalización como sigue:

$$Norm_2 : D \times H \rightarrow \mathbb{R}; \quad Norm_2(d, h) = \frac{V_{d,h}}{V_{\Delta(d),h}} = \rho_{d,h}. \quad (2.4)$$

Con este proceso, la posterior inversión de la normalización de los datos una vez predicha la proporción es muy simple: si $\hat{\rho}_{d,h}$ es la proporción predicha, entonces el valor de demanda de energía eléctrica predicha, $\hat{V}_{d,h}$, será el siguiente:

$$\hat{V}_{d,h} = \hat{\rho}_{d,h} * V_{\Delta(d),h},$$

resultado de despejar $V_{d,h}$ en la ecuación (2.4).

2.2.3. Tercera idea: normalización mixta

Esta tercera idea surge a raíz de que, si los datos han sido previamente normalizados mediante la primera idea, es decir, han sido transformados a un formato de día común, el proceso de la segunda idea de normalización es mucho más sencillo. En este caso, con todos los datos llevados a los esperados un día de entre semana completamente laborable, no hay que lidiar con las dos excepciones expuestas en la sección anterior.

Aparecen entonces dos posibles versiones de esta tercera idea: a la hora de hallar la proporción, se puede tomar el valor de la demanda normalizada de una semana atrás ó simplemente de un solo día atrás. En general, si n es el número de días atrás que se decide ir para tomar la proporción ($n = 7$ si se elige una semana ó $n = 1$ si se decide un solo día), se define la siguiente función, Δ_n , que hace el papel de Δ en la segunda idea de normalización:

$$\Delta_n : D \rightarrow D; \quad \Delta_n(d) = d - n.$$

De esta forma, se puede expresar la normalización como sigue:

$$Norm_3 : D \times H \rightarrow \mathbb{R}; \quad Norm_3(d, h) = \frac{V_{d,h} * C(d, h)}{V_{\Delta_n(d),h} * C(\Delta_n(d), h)} = \rho_{d,h}. \quad (2.5)$$

Nótese que el dividendo es el valor de la demanda del día a normalizar a la hora h , normalizado por el método de aproximación a día laborable. El divisor, por su parte, es el valor de la demanda del día con respecto al que se quiere tomar la proporción ($\Delta_n(d)$) a la hora h , también normalizado por el método de aproximación a día laborable.

Con este proceso, la posterior inversión de la normalización de los datos una vez predicha la proporción es muy simple: si $\hat{\rho}_{d,h}$ es la proporción predicha, entonces el valor de demanda de energía eléctrica predicha, $\hat{V}_{d,h}$, será el siguiente:

$$\hat{V}_{d,h} = \frac{\hat{\rho}_{d,h} * V_{\Delta_n(d),h} * C(\Delta_n(d), h)}{C(d, h)},$$

resultado de despejar $V_{d,h}$ en la ecuación (2.5). Al igual que se ha observado en la ecuación (2.5), se debe apreciar que en la desnormalización, el dividendo sería el valor normalizado por el método de aproximación a día laborable. Con ésto en cuenta, queda una desnormalización equivalente a la de dicho método.

MODELOS PRINCIPALES

Una vez observadas las características del sistema eléctrico español y de los datos históricos sobre la demanda de energía eléctrica en el capítulo anterior, es hora de presentar las herramientas que se van a usar para su previsión.

En todas ellas, se parte de una serie de N datos o muestras en forma de pares $\{(x, y)^i\}_{i=1}^N$. En este sentido, para cada par $(x, y)^i = (x^i, y^i)$, $x^i = (x_1^i, \dots, x_d^i)^t$ denota a las d variables regresoras (o de entrada) que servirán para tratar de predecir su variable respuesta (o de salida) asociada y^i . En el contexto concreto del problema de previsión de la demanda energética, a partir de ciertas variables regresoras que se explicarán con más detalle en el próximo capítulo, se predecirá el valor de demanda energética de un cierto día d a una cierta hora h , es decir, $y^i = V_{d,h}$.

Con esta idea en mente, el objetivo es encontrar una función $f : \mathbb{R}^d \rightarrow \mathbb{R}$ de tal forma que, para cada $i = 1, \dots, N$, $\hat{y}^i = f(x^i) \approx y^i$. Lo más habitual es seleccionar una cierta $f(x; W) \in \mathcal{F}$, donde \mathcal{F} es alguna familia de funciones paramétricas. Dentro de cada familia, se seleccionará la función, o más bien los parámetros W , que minimizan una cierta función de error $e(f(x; W)) = e(W)$. Habitualmente, esta función de error es el error cuadrático medio (*mean squared error*, ó *mse* para abreviar, en inglés).

Con tal fin, en este capítulo se presentan tres familias paramétricas distintas que serán usadas en la parte práctica del trabajo. Dichas explicaciones están basadas principalmente en lo expuesto en [6] y [7]. Por una parte, en la sección 3.1 se presentarán los modelos lineales, más concretamente el *Ridge Regressor*. Por otra parte, en la sección 3.2 se presentarán los modelos de redes neuronales, indagando en más profundidad en los conocidos como *Multilayer Perceptrons*. Para terminar, se presentarán los modelos basados en *Support Vector Machines (SVM)*, en la sección 3.3.

3.1. Métodos lineales

En general, los métodos lineales, a partir de una serie de variables regresoras (x_1, \dots, x_d) , predicen una variable respuesta (y) como combinación lineal de las variables regresoras. Es decir,

$$\hat{y} = \hat{y}(x_1, \dots, x_d; w_0, w_1, \dots, w_d) = w_0 + w_1 * x_1 + \dots + w_d * x_d.$$

Para simplificar, si se definen el vector de variables regresoras $x = (x_1, \dots, x_d)^t$ y el vector de parámetros $w = (w_1, \dots, w_d)^t$, $y \cdot$ denota el producto escalar de vectores, se tiene que

$$\hat{y} = \hat{y}(x; w_0, w) = w_0 + w \cdot x.$$

También, sin pérdida de generalidad, se puede suponer que tanto los vectores de variables regresoras como las variables respuesta están centradas en el origen, ya que éstas se pueden transformar para que cumplan la hipótesis restando su media. De esta manera se consigue que el peso de sesgo w_0 sea igual a 0, terminando con el modelo:

$$\hat{y} = \hat{y}(x; w) = w \cdot x.$$

En este momento es cuando entra en juego el problema de regresión: encontrar el w^* óptimo al problema. Para ello se buscará el w que minimice el error cuadrático medio dado el conjunto de observaciones o muestras que se posee. Es decir, se busca el $w^* = (w_1^*, \dots, w_d^*)^t$ tal que minimice

$$e_S(w^*) = \frac{1}{2N} \sum_{i=1}^N (w^* \cdot x^i - y^i)^2.$$

Sean X la matriz $N \times d$ e Y el vector $N \times 1$ que recogen la información de las observaciones o muestras como sigue:

$$X = \begin{pmatrix} x_1^1 & \cdots & x_d^1 \\ x_1^2 & \cdots & x_d^2 \\ \vdots & \ddots & \vdots \\ x_1^N & \cdots & x_d^N \end{pmatrix}, \quad Y = \begin{pmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{pmatrix}.$$

Entonces la fórmula del error cuadrático medio queda como sigue:

$$e_S(w) = \frac{1}{2N} (Xw - Y)^t (Xw - Y) = \frac{1}{2N} (w^t X^t X w - 2w^t X^t Y + Y^t Y).$$

De esta forma, encontrando el gradiente de esta última expresión e igualándolo a $\vec{0} = (0, \dots, 0)$ se obtendría un w^* que minimiza el error cuadrático medio, ya que en este caso, el error cuadrático medio es una forma cuadrática en w con el término asociado al cuadrado semidefinido positivo. Dicho gradiente queda como sigue:

$$\nabla e_S(w) = \frac{1}{N} X^t X w - \frac{1}{N} X^t Y.$$

Igualando a $\vec{0}$ y despejando w se obtiene finalmente el w^* que minimiza el error cuadrático medio:

$$w^* = (X^t X)^{-1} X^t Y.$$

En un principio, se podría pensar que ya que se ha resuelto el problema, pero puede surgir un nuevo inconveniente: $X^t X$ puede ser singular, es decir, podría ser que $X^t X$ no fuera invertible y, por tanto, no existiera $(X^t X)^{-1}$ y el w^* definido no tuviera sentido. Lo que sí que se puede afirmar sobre la matriz $X^t X$ es que es semidefinida positiva, es decir, sumándole una cantidad positiva de veces α la matriz identidad I , $X^t X + \alpha I$ es definida positiva y por tanto invertible. Con el objetivo de poder usar esta nueva matriz, se define un nuevo error cuadrático medio a minimizar como sigue:

$$e_S(w) = \frac{1}{2N} \sum_{i=1}^N (w \cdot x^i - y^i)^2 + \frac{\alpha}{2N} \|w\|^2, \quad (3.1)$$

que alcanza su mínimo en

$$w^* = (X^t X + \alpha I)^{-1} X^t Y.$$

Ésto es debido a que el gradiente de esta nueva medición de error es el siguiente:

$$\nabla e_S(w) = \frac{1}{N} X^t X w - \frac{1}{N} X^t Y + \frac{1}{N} \alpha I w,$$

y al igualar a $\vec{0}$ se obtiene dicho w^* . Aquí es donde entra en juego la *Regularización*. En vez de trabajar con $X^t X$ se usará la matriz $X^t X + \alpha I$, donde I es la matriz identidad de orden d y $\alpha > 0$, que como se ha expuesto anteriormente, sí se puede afirmar que es invertible.

Minimizar la expresión (3.1) es el conocido como problema de *Ridge Regression* [8]. Sin embargo, en otra ocasión se vuelve a dejar el trabajo a medias: se ha de encontrar el parámetro (más bien hiperparámetro) α que haga que su nuevo error cuadrático medio se minimice. Para ello, se realiza una validación cruzada:

- 1.– Se divide la muestra S en diversos conjuntos S_1, \dots, S_M de manera aleatoria. De esta manera se forman M pares (S_m, S_m^c) llamados *folds*.
- 2.– Se toman distintos valores posibles de los hiperparámetros, α en este caso, $\alpha_1, \dots, \alpha_L$.
- 3.– Para cada valor α_l , y para cada *fold* (S_m, S_m^c) , se crea un modelo con ese valor del hiperparámetro y con el conjunto S_m^c como muestra de entrenamiento y el conjunto S_m para medir su *score*.
- 4.– Se toma como valor de *score* del valor α_l la media de los resultados sobre todos los *folds*, y se elige como mejor valor del hiperparámetro el α_l que tenga el mejor resultado.

Este nuevo parámetro α regula el equilibrio varianza-sesgo. Volviendo a la ecuación (3.1) que define la medida de error del modelo de Ridge Regression, se puede ver el papel que tiene α en este sentido:

- 1.– Si α es demasiado grande, para minimizar la medida de error, es muy probable que w tienda a valores muy pequeños, cercanos a 0. De esta manera, el modelo tenderá a predecir mediante una constante, es decir, $\hat{y} = \hat{y}(x; w) = c$ con $c \in \mathbb{R}$ constante. Dicha constante tenderá entonces a la media muestral \bar{y} , que es la constante que minimiza el error dada la muestra. De esta manera, el modelo tendrá una varianza baja pero un sesgo que podría llegar a ser elevado, es decir, el modelo no depende en gran medida de los datos de entrenamiento pero puede llegar a tener un error en sus valores esperados alto.
- 2.– Si α es demasiado pequeño, w no tiene restricciones para “moverse” por todo \mathbb{R}^d , y poder adaptarse a la

muestra para minimizar su error muestral. De este modo, el sesgo del modelo será menor a cambio de una varianza que podría llegar a ser realmente alta, ya que w depende en gran medida de la muestra. Esta varianza se acentúa sobre todo cuando la matriz $X^t X$ es casi singular.

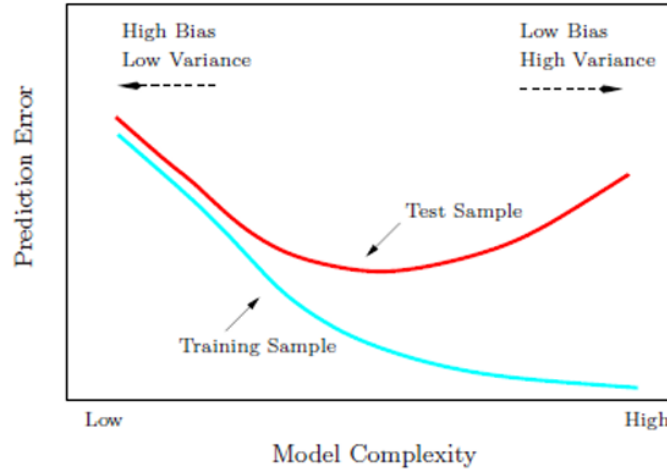


Figura 3.1: Error de predicción en función de la complejidad del modelo, tomada de [6]

Es decir, se ha de encontrar un α óptimo que reduzca lo suficiente el sesgo o la varianza sin aumentar demasiado el otro, ésto es, se ha de encontrar un equilibrio entre sesgo y varianza. La figura 3.1, tomada de [6], resume en gran medida esta idea, en un entorno idealizado. En el caso de Ridge Regression, en relación con lo expuesto anteriormente, un α grande implica un modelo más simple, que no se adapta bien ni a los datos con los que se ha entrenado ni a otros datos que puedan servir de test. Por otra parte, un α pequeño implica un modelo más complejo, capaz de adaptarse bien a los datos de entrenamiento, pero que, tal vez, no generaliza bien el problema.

Computacionalmente hablando, multiplicar e invertir matrices son operaciones muy costosas. En el problema de Ridge Regression, realizar la multiplicación $X^t X$ tiene un coste del orden de $O(N * d^2)$, e invertirla tiene un coste del orden de $O(d^3)$. Con el objetivo de reducir el coste computacional surge la idea del descenso por gradiente, un algoritmo iterativo para encontrar el w^* que minimiza el error cuadrático medio:

- 1.- Se parte de un $w^0 \in \mathbb{R}^d$ aleatorio.
- 2.- Se calcula iterativamente w^{k+1} a partir de w^k . Para ello, se toma un *mini-batch* B de n_B muestras de las N totales, cuya información queda recogida en la matriz X_B de tamaño $n_B \times d$ y el vector Y_B de tamaño $n_B \times 1$ de igual manera que hacían las matrices X e Y con la muestra total. Cada iteración es como sigue:

$$w^{k+1} = w^k - \rho \nabla e_B(w^k) = w^k - \frac{\rho}{n_B} (X_B^t X_B w^k - X_B^t Y_B + \alpha I w),$$

donde $\rho > 0$ es la *learning rate*, un nuevo hiperparámetro que determina la velocidad y precisión de la convergencia.

Obsérvese que, si se tiene $\lim_{k \rightarrow \infty} w^k = w^*$, entonces se tiene $\nabla e_B(w^*) = 0$. Suponiendo que el *mini-batch* B representa en gran medida la muestra total, w^* es un mínimo. Además, este método reduce el coste computacional al no calcularse ninguna inversa de matrices y las matrices que se multiplican son de una dimensión menor.

3.2. Redes neuronales

La idea detrás de las redes neuronales es el funcionamiento de las neuronas [9]. Éstas, a partir de impulsos de entrada que reciben de otras neuronas, envían, o no, unos impulsos de salida. Este funcionamiento se intenta transformar en un estimador que, al igual que en los métodos lineales, a partir de una serie de variables regresoras (x_1, \dots, x_d) predice una variable respuesta (y) , mediante:

- 1.– Una capa de neuronas inicial de entrada, que recibe como impulso de entrada las variables regresoras x_1, \dots, x_d .
- 2.– Una serie de capas ocultas de neuronas. Cada una de ellas recibe como impulsos de entrada los impulsos de salida de la capa anterior.
- 3.– Una capa de salida.

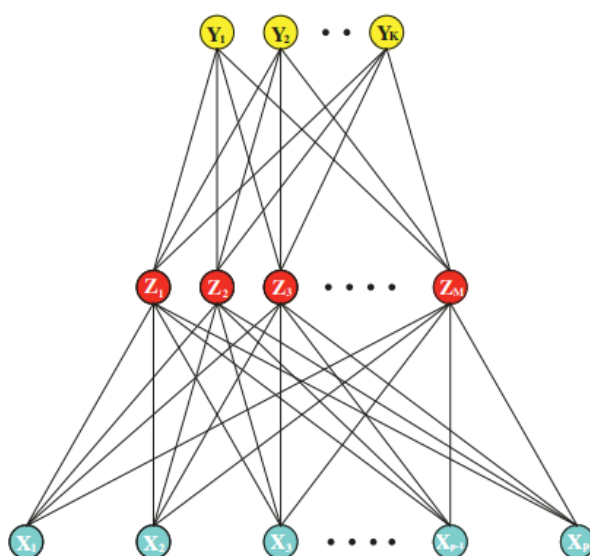


Figura 3.2: Arquitectura de una red neuronal, tomada de [6]

En la figura 3.2 se puede observar un esquema de la arquitectura de una red neuronal, con los elementos descritos con anterioridad, tomada de [6]. Como se puede apreciar, cada neurona de cada capa recibe por un lado los posibles impulsos de todas las neuronas de la capa anterior y, en caso de activarse, envía su impulso de salida a todas las neuronas de la capa siguiente. Es decir, cada neurona está conectada con todas las neuronas, tanto de la capa anterior como de la posterior.

Dichas conexiones representan combinaciones lineales de los impulsos que recibe cada neurona. Sin embargo, las activaciones de cada neurona se modelizan mediante funciones posiblemente no lineales. De esta manera, con la superposición de distintas capas, se consigue un resultado potencialmente no lineal. Para formalizar estas explicaciones se definen los siguientes valores:

- o_j^h : el valor de salida de la neurona j de la capa h .
- a_j^h : el valor de activación de la neurona j de la capa h .
- w_{ij}^h : el peso de la salida de la neurona j de la capa $h - 1$ para la activación de la neurona i de la capa h .
- b_j^h : el valor de sesgo de la neurona j de la capa h .

- n_h : el número de neuronas de la capa h .

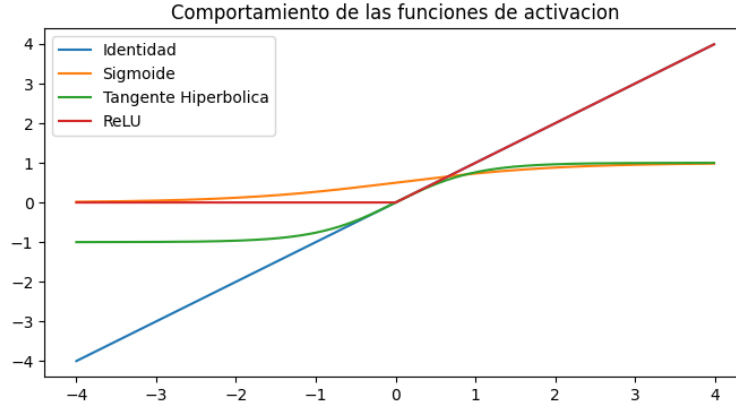


Figura 3.3: Comportamiento de las funciones de activación

La capa $h = 0$ es la capa de entrada. A partir de estas definiciones se tienen las igualdades $a_i^h = \sum_{j=1}^{n_{h-1}} w_{ij}^h o_j^{h-1} + b_i^h$ y $o_i^h = \varphi(a_i^h)$, donde $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ es una función de activación. En forma matricial, si se define una nueva matriz W^h de dimensión $n_h \times n_{h-1}$, cuyas componentes son los pesos w_{ij}^h , y un nuevo vector b^h de dimensión $n_h \times 1$, cuyas componentes son los valores de sesgo b_j^h , se tienen:

$$a^h = W^h o^{h-1} + b^h, \quad o^h = \varphi(a^h).$$

La función de activación $\varphi(\cdot)$ se suele tomar de entre las siguientes opciones, cuyos comportamientos se pueden apreciar en la figura 3.3:

- La función identidad: $\varphi(x) = x$.
- La función sigmoide: $\varphi(x) = \frac{1}{1+e^{-x}}$.
- La función tangente hiperbólica: $\varphi(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.
- La función *ReLU*: $\varphi(x) = \max(0, x)$.

En el caso del problema de regresión, lo habitual es tomar la función *ReLU* como función de activación de las capas ocultas y la función identidad en la capa de salida.

En cierto sentido, se puede entender los modelos lineales como redes neuronales muy simples, sin ninguna capa oculta. Cada neurona de la capa inicial envía como impulso a la capa final un valor de una de las variables regresoras. La capa final los recibe y realiza la combinación lineal con los pesos de cada conexión. La función de activación es en este caso la función identidad.

En el caso de las redes neuronales se tiene que el valor predicho es de la forma:

$$\hat{y} = \hat{y}(x; W^1, b^1, \dots, W^H, b^H).$$

En este sentido, se buscan los parámetros $W^1, b^1, \dots, W^H, b^H$ que minimicen el error cuadrático

medio:

$$\begin{aligned} e_S(W^1, b^1, \dots, W^H, b^H) &= \frac{1}{2} \mathbb{E}_{x,y} \left[(y - \hat{y}(x; W^1, b^1, \dots, W^H, b^H))^2 \right] \\ &= \mathbb{E}_{x,y} \left[e^l(x, y; W^1, b^1, \dots, W^H, b^H) \right] \\ &= \int e^l(x, y; W^1, b^1, \dots, W^H, b^H) p(x, y) dx dy, \end{aligned}$$

donde $e^l(x, y; W^1, b^1, \dots, W^H, b^H)$ denota el error cuadrático local:

$$e^l(x, y; W^1, b^1, \dots, W^H, b^H) = \frac{1}{2} (y - \hat{y}(x; W^1, b^1, \dots, W^H, b^H))^2.$$

Para ello se calcula el gradiente del error cuadrático local. Primero ha de observarse que

$$\frac{\partial a_i^h}{\partial w_{ij}^h} = o_j^{h-1}, \quad \frac{\partial o_i^h}{\partial a_i^h} = \varphi'(a_i^h).$$

Por otra parte, la regla de la cadena es también clave, en el sentido de que

$$\frac{\partial e^l}{\partial w_{ij}^h} = \frac{\partial e^l}{\partial a_i^h} \frac{\partial a_i^h}{\partial w_{ij}^h} = \frac{\partial e^l}{\partial a_i^h} o_j^{h-1} = \delta_i^h o_j^{h-1}.$$

Tomando como función de activación en la capa de salida la función identidad, se tiene que $\hat{y} = o^H = a^H$, luego

$$\delta^H = \frac{\partial e^l}{\partial a^H} = \frac{\partial e^l}{\partial \hat{y}} = \hat{y} - y,$$

y por tanto,

$$\frac{\partial e^l}{\partial w_j^H} = (\hat{y} - y) o_j^{H-1}.$$

A partir de este valor, se pueden ir calculando los valores hacia capas anteriores como sigue. Supóngase que se están calculando los valores del gradiente para la capa h , y que ya se han calculado para las capas $h+1, \dots, H$. Entonces:

$$\frac{\partial e^l}{\partial w_{ih}^h} = \frac{\partial e^l}{\partial a_i^h} o_j^{h-1},$$

y por otra parte,

$$\frac{\partial e^l}{\partial a_i^h} = \sum_{k=1}^{n_{h+1}} \left(\frac{\partial e^l}{\partial a_k^{h+1}} \frac{\partial a_k^{h+1}}{\partial a_i^h} \right) = \sum_{k=1}^{n_{h+1}} \left(\delta_k^{h+1} \frac{\partial a_k^{h+1}}{\partial o_i^h} \frac{\partial o_i^h}{\partial a_i^h} \right) = \varphi'(a_i^h) \left[\sum_{k=1}^{n_{h+1}} \delta_k^{h+1} w_{ki}^{h+1} \right].$$

Al igual que ocurría con los métodos lineales, es necesario tener un equilibrio entre la varianza y el sesgo del estimador. Con el objetivo de lograrlo, se introduce la *Regularización* también en el caso de las redes neuronales. En ese sentido, para controlar el dominio en el que puede tomar valores

$\mathcal{W} = (W^1, \dots, W^H)^t$, se introduce a la medida de error un término que tenga en cuenta su norma. De esta manera, definiendo el vector de sesgos $\mathcal{B} = (b^1, \dots, b^H)$, el nuevo error a minimizar queda:

$$e_R(\mathcal{W}, \mathcal{B}) = e_S(\mathcal{W}, \mathcal{B}) + \frac{\lambda}{2} \|\mathcal{W}\|^2,$$

quedando entonces su gradiente como:

$$\nabla_{\mathcal{W}} e_R(\mathcal{W}, \mathcal{B}) = \nabla_{\mathcal{W}} e_S(\mathcal{W}, \mathcal{B}) + \lambda \mathcal{W}.$$

En relación con lo observado para el parámetro α de *Ridge Regression* y la figura 3.1, un λ grande restringe los posibles valores de los pesos W^1, \dots, W^H , reduciendo la complejidad del modelo y de este modo obteniendo un estimador con una varianza menor pero con un error de sesgo potencialmente alto, que haga que el estimador no se ajuste bien a los datos de la muestra y tampoco generalice el problema.

Por su parte, para un valor bajo del hiperparámetro λ , los pesos tienen una menor restricción sobre los posibles valores a tomar, aumentando la complejidad del modelo, creando un estimador que se ajuste muy bien a la muestra, pero que no sea capaz de generalizar. En ese sentido, se tendría un error de sesgo pequeño, pero una gran varianza. Sin embargo, en el caso de las redes neuronales, λ no es el único valor que influye en la complejidad del modelo. El número de capas ocultas, y el número de neuronas en cada una de ellas, también juegan un papel crucial en ese sentido. Que una red neuronal tenga más capas ocultas y más neuronas en cada una de ellas que otra red neuronal, implica que la primera tiene más pesos, y por tanto, que sea más compleja, con las implicaciones que ésto tiene sobre la varianza y el sesgo, y la capacidad de generalizar el problema, ya comentadas con anterioridad.

Por este motivo, se ha de realizar una hiperparametrización, tanto del valor de λ como de la cantidad de capas ocultas y neuronas. Para ello se realiza un proceso de validación cruzada, equivalente al que se realiza para el problema de *Ridge Regression*, con estas nuevas cantidades.

Por último, al igual que para el caso de *Ridge Regression*, en las redes neuronales también se usa el algoritmo de descenso por gradiente para la búsqueda de los pesos $\mathcal{W} = W^1, b^1, \dots, W^H, b^H$ óptimos, con el objetivo de reducir el coste computacional. En concreto, lo más habitual es hacer uso del algoritmo *Adam* [10]. En él, para cada iteración k , se realizan los siguientes pasos:

- 1.– Se toma un *mini-batch* para calcular el valor del gradiente en \mathcal{W}_{k-1} , que se denotará por g_k . Se define también $g_k^2 = g_k \cdot g_k$.
- 2.– Se actualizan las medias suavizadas exponencialmente de g_k y g_k^2 :
 - $m_k = \beta_1 m_{k-1} + (1 - \beta_1) g_k$;
 - $v_k = \beta_2 v_{k-1} + (1 - \beta_2) g_k^2$.
- 3.– Se calculan las correcciones de sesgo:
 - $\hat{m}_k = \frac{1}{1 - \beta_1} m_k$;

$$\bullet \hat{v}_k = \frac{1}{1-\beta_2} v_k.$$

4.– Se actualizan los pesos:

$$\mathcal{W}_k = \mathcal{W}_{k-1} - \alpha \frac{\hat{m}_k}{\sqrt{\hat{v}_k} + \epsilon}.$$

Para la realización de dichos cálculos, se suelen tomar como valores estándar $\alpha = 0,001$, $\beta_1 = 0,9$, $\beta_2 = 0,999$ y $\epsilon = 10^{-8}$, tal y como se recomienda en [10].

3.3. Support Vector Machines

Las *Support Vector Machines* fueron propuestas inicialmente para problemas de clasificación. Nacieron a partir de una evolución de los métodos lineales, en los que un hiperplano separa dos regiones sobre las que se clasifican nuevas observaciones. Para su entrenamiento, se minimiza el número de muestras clasificadas de forma incorrecta. La mejora de las *SVM* radica en no penalizar las muestras clasificadas incorrectamente que se encuentran cercanas al hiperplano que separa las regiones [11] [12].

Trasladando ésto al problema de regresión, al igual que para clasificación, los *Support Vector Regressors* se basan en la teoría de métodos lineales. En este sentido se parte de un modelo lineal:

$$\hat{y} = \hat{y}(x; w_0, w) = w_0 + w \cdot x.$$

La función de error a minimizar, con la regularización ya aplicada mediante el hiperparámetro $C = \frac{N}{2\alpha}$, es la siguiente:

$$e_S(w, w_0) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N [y^i - (w \cdot x^i + w_0)]_{\epsilon},$$

que hace uso de la función de insensibilidad ϵ :

$$[z]_{\epsilon} = \max(0, |z| - \epsilon).$$

Esta función de insensibilidad hace que no se penalicen errores de predicción menores a ϵ . Nótese también que ϵ es un nuevo hiperparámetro del modelo. Dicho problema de minimización es complicado de resolver, ya que la función de error no es diferenciable. Aun así, el problema inicial se puede transformar en un nuevo problema de minimización. Se define una nueva función de error, equivalente, como sigue:

$$e_S(w, w_0, \xi, \eta) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \eta_i),$$

con las siguientes restricciones, para todo $i = 1, \dots, N$:

$$\begin{aligned} -\xi_i - \epsilon &\leq w \cdot x^i + w_0 - y^i; \\ \eta_i + \epsilon &\geq w \cdot x^i + w_0 - y^i; \\ \xi_i, \eta_i &\geq 0. \end{aligned}$$

Minimizar e_S con estas restricciones es el problema primal. El primer paso para su resolución es desarrollar su lagrangiano, para finalmente definir una función dual más sencilla:

$$\begin{aligned} L(w, w_0, \xi, \eta, \alpha, \beta, \gamma, \delta) = & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \eta_i) \\ & - \sum_{i=1}^N \alpha_i (w \cdot x^i + w_0 - y^i + \xi_i + \epsilon) \\ & + \sum_{i=1}^N \beta_i (w \cdot x^i + w_0 - y^i - \eta_i - \epsilon) \\ & - \sum_{i=1}^N \gamma_i \xi_i - \sum_{i=1}^N \delta_i \eta_i. \end{aligned}$$

Se define entonces la función dual $\Theta(\alpha, \beta, \gamma, \delta) = \min_{w, w_0, \xi, \eta} L(w, w_0, \xi, \eta, \alpha, \beta, \gamma, \delta)$. Por construcción, se tiene la siguiente relación:

$$\Theta(\alpha, \beta, \gamma, \delta) \leq L(w, w_0, \xi, \eta, \alpha, \beta, \gamma, \delta) \leq e_S(w, w_0, \xi, \eta).$$

Derivando con respecto a w , w_0 , ξ y η , e igualando a 0 para encontrar un mínimo en dichos parámetros, se obtienen los siguientes resultados:

- De $\nabla_w L = 0$ y $\frac{\partial L}{\partial w_0} = 0$ se obtiene:

$$w = \sum_{i=1}^N \alpha_i x^i - \sum_{i=1}^N \beta_i x^i; \quad \sum_{i=1}^N \alpha_i = \sum_{i=1}^N \beta_i.$$

- Y de $\frac{\partial L}{\partial \xi_i} = 0$ y $\frac{\partial L}{\partial \eta_i} = 0$ se obtiene que para todo $i = 1, \dots, N$:

$$C = \alpha_i + \gamma_i; \quad C = \beta_i + \delta_i.$$

Introduciendo estos resultados en el lagrangiano, éste se puede simplificar, llegando finalmente a la expresión de la función dual Θ :

$$\begin{aligned} \Theta(\alpha, \beta, \gamma, \delta) = & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i - \beta_i)(\alpha_j - \beta_j) x^i \cdot x^j \\ & + \epsilon \sum_{i=1}^N (\alpha_i + \beta_i) - \sum_{i=1}^N y^i (\alpha_i - \beta_i), \end{aligned}$$

con las siguientes restricciones,

$$0 \leq \alpha_i, \beta_i \leq C \text{ para todo } i = 1, \dots, N;$$

$$\sum_{i=1}^N \alpha_i = \sum_{i=1}^N \beta_i.$$

Obsérvese cómo Θ no es función ni de γ ni de δ . De esta manera, el problema dual se resuelve encontrando $\min_{\alpha, \beta} \Theta(\alpha, \beta)$ entre los valores de α y β que cumplen las restricciones anteriores.

Se puede demostrar que, si $(w^*, w_0^*, \xi^*, \eta^*)$ es la solución óptima del problema primal y (α^*, β^*) es la solución óptima del problema dual, entonces se tiene que:

$$e_S(w^*, w_0^*, \xi^*, \eta^*) = \Theta(\alpha^*, \beta^*). \quad (3.2)$$

Por este motivo, el problema a resolver será el dual, y para ello, se hace uso del algoritmo *SMO* (*Sequential Minimal Optimization*). La explicación de este algoritmo sale fuera del alcance de este trabajo. Consúltese [13] para más información sobre *SMO*.

De la ecuación (3.2) se obtienen las siguientes condiciones de optimalidad. Para todo $i = 1, \dots, N$:

$$\begin{cases} 0 &= \alpha_i^*(w^* \cdot x^i + w_0^* - y^i + \xi_i^* + \epsilon); \\ 0 &= \beta_i^*(w^* \cdot x^i + w_0^* - y^i - \eta_i^* - \epsilon); \\ 0 &= (C - \alpha_i^*)\xi_i^*; \\ 0 &= (C - \beta_i^*)\eta_i^*. \end{cases}$$

A partir de estas condiciones, se obtiene que, si $0 < \alpha_p^* < C$, entonces $\xi_p^* = 0$ y $w^* \cdot x^p + w_0^* - y^p = -\epsilon$. De forma similar, si $0 < \beta_q^* < C$, entonces $\eta_q^* = 0$ y $w^* \cdot x^q + w_0^* - y^q = \epsilon$. Dichos datos de la muestra x^p, x^q son los vectores soporte del modelo. Nótese que definen un tubo de radio ϵ .

Por otra parte, también se tiene que, si $\xi_i^* > 0$, entonces $\alpha_i^* = C$. De forma equivalente, si $\eta_i^* > 0$, entonces $\beta_i^* = C$. Despejando de las ecuaciones anteriores, también se puede obtener el w_0^* óptimo: si (x^p, y^p) es un dato tal que $0 < \alpha_p^* < C$ entonces $w_0 = -\epsilon - w^* \cdot x^p + y^p$. De esta manera, junto al w^* óptimo encontrado con anterioridad a partir de $\nabla_w L = 0$, el modelo queda:

$$\hat{y} = \hat{y}(x; w_0^*, \alpha^*, \beta^*) = w_0^* + \sum_{i=1}^N (\alpha_i^* - \beta_i^*) x^i \cdot x.$$

Es aquí cuando entra en juego el punto que diferencia las *SVM* de los métodos lineales: el *Kernel Trick*. Según el desarrollo realizado hasta el momento, el modelo solamente aplica transformaciones lineales. Sin embargo, este hecho se puede cambiar si se toma una aplicación $\Phi(x)$ que lleve los puntos de la muestra a un espacio de dimensión potencialmente infinita. Además, como sobre los puntos solamente se calculan productos escalares, de dicho espacio solamente se necesita conocer el producto escalar, al que se llamará como *kernel*, $k(x^i, x) = \Phi(x^i) \cdot \Phi(x)$. De entre todas las posibles elecciones de *kernel* destaca el *gaussiano*:

$$k(x^i, x) = e^{-\gamma \|x^i - x\|^2}.$$

Nótese que, al añadir el *kernel gaussiano* al modelo, se le añade un nuevo hiperparámetro γ . De esta forma, el modelo final queda:

$$\begin{aligned}\hat{y} = \hat{y}(x; w_0^*, \alpha^*, \beta^*) &= w_0^* + \sum_{i=1}^N (\alpha_i^* - \beta_i^*) \Phi(x^i) \cdot \Phi(x) \\ &= w_0^* + \sum_{i=1}^N (\alpha_i^* - \beta_i^*) k(x^i, x) \\ &= w_0^* + \sum_{i=1}^N (\alpha_i^* - \beta_i^*) e^{-\gamma \|x^i - x\|^2}.\end{aligned}$$

En el caso de las *SVR*, la complejidad del modelo la rigen los hiperparámetros C , ϵ y γ . El hiperparámetro C juega en este caso un papel similar a α en *Ridge Regression* o redes neuronales. En ese sentido, un C grande implica que los pesos pueden tomar valores en un espacio mayor, ajustándose mejor a los datos de entrenamiento, generando un modelo más complejo, pero resultando en una potencial pérdida de generalidad. Por su parte, un C pequeño implica que el modelo tiende a una constante, con bajos resultados tanto en la muestra de entrenamiento como en la posterior generalización.

Por su parte, el hiperparámetro γ juega un papel similar regularizando el *kernel gaussiano*. Un γ grande implica un modelo complejo, y un γ pequeño un modelo más sencillo, con las implicaciones que ésto tiene sobre los resultados. Finalmente, el hiperparámetro ϵ regula la penalización por el error cometido en la predicción. En ese sentido, un ϵ grande no penaliza los errores, derivando en un modelo sencillo. Por su parte, un ϵ pequeño sí que los penaliza, generando un modelo complejo.

EXPERIMENTOS

Una vez vistos los métodos de aprendizaje automático, llega el momento de ponerlos en uso sobre datos reales. Dichos datos serán los valores de demanda de energía eléctrica presentados con anterioridad. Aun así, los experimentos a realizar no se componen simplemente de probar esas herramientas. Antes que nada hay que elegir el entorno sobre el que se va a trabajar, entendiendo las necesidades del problema a tratar. En ese sentido, se presentará en la sección 4.1 el entorno elegido en este caso, junto a las razones de dicha decisión.

Para llevar a cabo los experimentos, otra de las piezas fundamentales son los propios datos. Dichos datos hay que obtenerlos, y tratarlos, antes de poder trabajar con ellos. Por este motivo, en la sección 4.2 se explican los procedimientos seguidos para su obtención y procesado. Una vez tratados, se debe analizar si la nueva información obtenida se corresponde a la que se esperaba obtener, analizando cómo puede afectar dicha transformación a los resultados finales de los experimentos. Se puede observar el análisis de los datos normalizados para este trabajo en la sección 4.3.

Por último, una vez realizadas todas las tareas anteriores, se procederá a explicar los experimentos propiamente dichos. En la sección 4.4, se explicará la metodología seguida para su realización, junto a un análisis de los resultados obtenidos, dividido por cada uno de los métodos de aprendizaje automático usados. Para la consulta y descarga del código desarrollado se puede acudir al siguiente repositorio de *GitHub*: <https://github.com/CarlosAliagaSamper/TFG-Informatica>.

4.1. Descripción del entorno

Como principal lenguaje de programación para el desarrollo de los experimentos se decidió usar *Python*. Esta decisión se realizó basándose en la gran cantidad y calidad de librerías ya desarrolladas en este lenguaje para el análisis de datos y aprendizaje automático. Entre ellas, cabe destacar la librería *sklearn* [14]. En ella, se encuentra ya desarrollada toda la funcionalidad para la creación y uso de todos los estimadores presentados en el capítulo anterior, desde su hiperparametrización y entrenamiento hasta su posterior uso para predecir nuevos valores, estimando su precisión. Dentro de *sklearn*, cabe destacar las clases *KFold* y *GridSearchCV* usadas para la hiperparametrización en este

trabajo, y las clases *Ridge*, *MLPRegressor* y *SVR* usadas para la creación de los modelos. Además, para poder utilizar los modelos de *SVR* en contextos de más de una variable de salida, se puede hacer uso también de la clase *MultiOutputRegressor*.

A la hora de la obtención de los datos, *Python* también contaba con una gran ventaja sobre el resto de lenguajes de programación. Gracias a la existencia de la librería *requests* [15], la obtención de los datos sobre demanda de energía eléctrica se simplificaba, gracias a la facilidad que esta librería aportaba a la hora de realizar peticiones a la api de e-sios. También, para la obtención de los datos de temperatura, *Python* también cuenta con la librería *meteostat* [16], que comparte a través de una api muy sencilla diferentes datos meteorológicos recogidos a lo largo del tiempo en distintas estaciones meteorológicas de todo el planeta. Además, para el posterior estudio de los datos obtenidos, *Python* cuenta con librerías tanto para su agrupación, tratado y análisis (*pandas* [17] o *numpy* [18]), como para su visualización mediante gráficos (*matplotlib* [19] o *hvplot* [20]). Finalmente, *Python* también cuenta con el paquete *scipy* [21], con la función *ranksums()*, la cual se usará para la comparación de modelos.

Una parte fundamental de los experimentos es conseguir, y tratar si es necesario, los datos con los que se van a trabajar. En la siguiente sección se presenta el proceso seguido para la obtención de los datos, tanto de demanda de energía eléctrica en España como de temperatura. También se expondrá cómo han sido tratados.

4.2. Obtención y procesado de los datos

Los datos de demanda de energía eléctrica se consiguen directamente de la plataforma e-sios, donde Red Eléctrica de España publica una larga lista de datos relacionados con la generación y consumo de energía eléctrica en España. Los valores obtenidos están medidos en MWh, pero fueron transformados a GWh para facilitar su uso. Por su parte, los datos de temperatura se consiguen mediante *meteostat*, una librería de *Python* con una api que comparte datos meteorológicos de distintas estaciones. Los datos de temperatura obtenidos están medidos en grados Celsius (°C).

Una vez obtenidos los datos, éstos son juntados y guardados, asociados a su día y hora (en horario UTC) para su posterior tratamiento y uso. Por una parte, los datos de demanda energética siguen los procesos de normalización expuestos en la sección 2.2. Para ello, se tomaron los calendarios laborales de las distintas provincias de España durante los años 2021, 2022 y 2023 de [22] y la población de las mismas de Wikipedia [23].

Por otra parte, los datos de temperatura se han tratado buscando una *insensibilidad térmica*. Se ha buscado centrar los valores de la temperatura debido a que es natural pensar que, a partir de un cierto valor en el que la población no siente el suficiente frío como para encender la calefacción, pero tampoco el suficiente calor como para encender el aire acondicionado, la diferencia en la temperatura, ya sea hacia arriba o hacia abajo, afecta de la misma manera a la demanda de energía eléctrica.

Para encontrar dicho centro, como si de un hiperparámetro se tratara, se fueron probando distintos valores en una rejilla y midiendo su correlación con los datos de demanda energética, tanto sin normalizar como normalizados por los distintos métodos expuestos con anterioridad. En cada caso se fue midiendo la correlación de los datos de demanda energética y la temperatura centrada. Los resultados se pueden observar en la figura A.1 adjunta en el anexo A.

Normalización	T (°C)
Sin normalización	13,0
Día laborable	13,0
Semana Previa	16,0
Mixta 7 días	17,0
Mixta 1 día	17,0

Tabla 4.1: Temperaturas centrales en función de la normalización de los datos

Con estos resultados se decidió centrar los datos de temperatura mediante la fórmula siguiente:

$$t_{d,h}^c = |t_{d,h} - T|^2,$$

donde $t_{d,h}^c$ y $t_{d,h}$ representan los valores de la temperatura centrada y sin centrar respectivamente del día d a la hora h . Por su parte, T es la temperatura central, que varía dependiendo de la normalización de los datos asociados. Se pueden consultar estos valores en la tabla 4.1. Por otro lado, se decidió elevar al cuadrado la temperatura centrada ya que, tras probar varios exponentes, éste era el valor que mayor correlación conseguía entre los datos de temperatura centrada y los datos de demanda de energía eléctrica.

Una vez tratados los datos es momento de analizarlos, y compararlos con los datos originales, con el objetivo de comprobar si se han conseguido los resultados que se deseaban. En ese sentido, se presenta el análisis de los datos normalizados en la siguiente sección.

4.3. Análisis de los datos normalizados

Se comenzará observando los resultados de la normalización de los datos de demanda energética. Con respecto al primer proceso de normalización, el de aproximación a día laborable, como su nombre indica, se debería de poder observar cómo los datos de cualquier día, ya sea sábado, domingo o festivo, se asemejan a los de un día de entre semana. Este hecho se puede apreciar claramente en los histogramas que muestra la figura A.2 adjunta en el anexo A: en los histogramas superiores, correspondientes a los datos sin normalizar se puede apreciar cómo, para cada día seleccionado, los valores no son comparables, mientras que en los histogramas inferiores, tras normalizar, los datos tienen patrones más similares.

Por otra parte, comparando los histogramas de la figura A.3 adjunta en el anexo A, se puede apreciar que, como cabría esperar, los datos globales de demanda energética crecen al normalizarlos por aproximación a día laborable (se han desplazado hacia la derecha). Además, estos nuevos datos normalizados siguen un patrón comparable al que se puede apreciar en los datos sin normalizar de los martes en el primer histograma de la figura A.2. Por su parte, los datos normalizados por semana previa y por el método mixto, tanto con 7 días atrás como con 1, deben permanecer cercanos a 1. En los histogramas de la figura A.4 adjunta en el anexo A también se puede apreciar la consecución de dicho objetivo en los tres casos anteriores.

Por otro lado, con respecto a los datos de temperatura, se puede hablar de la mejora que se espera que surja a raíz de utilizar la temperatura centrada sustituyendo a la temperatura sin centrar. Volviendo a la figura A.1, se puede observar cómo, para los casos de los datos de demanda energética sin normalizar y con normalización por aproximación a día laborable, la correlación base es considerablemente mayor que para los otros tres métodos de normalización. De esta manera, el incremento en la correlación de entorno a 0,1 para los datos sin normalizar, representa una mejora del 50 % ya que se parte de una base de 0,2. Por su parte, para los datos normalizados por aproximación a día laborable, que también partían de una correlación base de 0,2, se consigue una mejora del 75 %, llegando a una correlación de 0,35, es decir, 0,15 más que la base.

En ese sentido, no se espera una gran mejora en los resultados de las predicciones con respecto a los resultados de la temperatura sin centrar en estos dos casos. Sin embargo, para las normalizaciones por semana previa y mixta sí que se nota una subida más significativa, ya que parten de una base más baja. El aumento en la correlación que tienen los datos de la demanda energética normalizados por semana previa y por normalización mixta de 7 días son de entorno a 0,27, representando una mejora del 440 %. Por su parte, la correlación base de los datos de demanda energética con normalización mixta de 1 día era tan cercana a 0 que también se debería apreciar una mejora en los resultados.

Una vez los datos tratados han sido analizados, se procede a realizar los experimentos propiamente dichos sobre las herramientas explicadas en el capítulo anterior. A continuación, se explicará la metodología general seguida en la realización de los experimentos, indagando en las particularidades y resultados de *Ridge Regression* (sección 4.4.1), *Perceptrones Multicapa* (sección 4.4.2) y *SVR* (sección 4.4.3).

4.4. Experimentos

Para cada modelo la idea será la misma. Se fijará un número n de horas atrás que servirán para predecir otra cantidad fijada m de horas adelante (horizontes). En este trabajo se decidió tomar $n = 12$ horas atrás y $m = 6$ horas adelante. Además, se incluirán en el conjunto de variables de entrada de cada modelo la información de temperatura.

El primer paso a realizar será separar el conjunto total de datos en un primer subconjunto para hiperparametrizar y entrenar cada uno de los modelos y un segundo subconjunto para realizar un test y valorar cada uno de los modelos. Este proceso se realizará tanto para los datos sin normalizar como para cada tipo de normalización propuesta en el capítulo 2. Además, se valorará en qué medida aporta información y precisión a la hora de predecir la demanda la inclusión de los datos sobre temperatura, tanto centrada como sin centrar (se usarán las siglas ST, CT y CTC correspondientes a Sin Temperatura, Con Temperatura y Con Temperatura Centrada en las tablas con los resultados).

Por una parte, el conjunto de *training* estará conformado por los datos de los dos primeros años, 2021 y 2022. Éstos serán a su vez divididos en otros dos subconjuntos para poder realizar una validación cruzada (*cross validation* en inglés) a la hora de hiperparametrizar, el primero conformado, aproximadamente, por los datos de 2021 y el segundo, aproximadamente, por los de 2022. Por otra parte, el conjunto de *test* estará conformado por los datos del tercer y último año, 2023.

El segundo paso a realizar, una vez separados los datos, será la hiperparametrización. Como se ha visto en el capítulo anterior, cada modelo posee una serie de hiperparámetros que sirven para entrenarlos. Se ha de elegir los valores óptimos para cada uno de ellos. El proceso de búsqueda de estos valores óptimos es la hiperparametrización. Como se ha explicado con anterioridad, para este paso se utiliza el conjunto de *training* y se realizará mediante validación cruzada, es decir, para cada valor de los hiperparámetros, se entrena con uno de los subconjuntos de entrenamiento y se mide la precisión con el otro, y viceversa.

Una vez obtenidos los mejores estimadores de cada modelo para cada caso se realizará el tercer y último paso: la medición de los *scores* sobre el conjunto de test, es decir, una valoración de cuán bien realizan la previsión de demanda energética cada uno de los estimadores. Para ello se usa como medida el error absoluto medio (*mean absolute error*, ó *MAE* para abreviar, en inglés) sobre los datos previstos tras desnormalizar. A pesar de la existencia de diversas medidas para el error (como por ejemplo el coeficiente R^2) se ha decidido usar *MAE* al ser éste el más habitual en el mundo eléctrico, ya que da información sobre el error que se espera cometer en las mismas unidades de energía.

En las siguientes secciones se explicarán las particularidades de los experimentos para cada modelo utilizado: *Ridge Regression*, *Perceptrones Multicapa* y *SVR*. También se compararán sus resultados con los de una previsión por persistencia, es decir, prediciendo el valor de demanda de las horas adelante como la demanda medida en la hora actual, y con los de la previsión realizada por Red Eléctrica de España. Éste último resultado solamente se comparará con nuestra previsión de la primera hora adelante, ya que REE revisa y actualiza cada hora su previsión de demanda de energía eléctrica que comparte mediante la plataforma e-sios.

Dicho valor del *MAE* cometido por la previsión de REE es de 0,234 GWh. Aun así, el error cometido por REE debe considerarse como un valor base, no comparable a los resultados obtenidos en este trabajo, ya que REE va actualizando sus modelos a lo largo del tiempo, mientras que este trabajo

Horizonte	1	2	3	4	5	6
MAE (GWh)	0,941	1,771	2,491	3,088	3,576	3,950

Tabla 4.2: MAE de la previsión por persistencia (GWh)

hace uso de un único modelo para todas las predicciones del año. Por otra parte, los resultados de la previsión por persistencia se pueden observar en la tabla 4.2.

4.4.1. Experimentos sobre Ridge Regression

El modelo de *Ridge Regression* es un modelo lineal muy simple: tan solo posee un hiperparámetro, α . Para su hiperparametrización se ha usado una rejilla con valores de la forma 10^k para $k = -8, \dots, 3$. Los CV MAE de la hiperparametrización de cada uno de los casos pueden observarse en las figuras B.1, B.2 y B.3 adjuntas en el anexo B. Nótese que, a pesar de la diferencia de escalas en los distintos casos, el comportamiento es similar en todos ellos: se encuentran unos valores del CV MAE estables hasta en torno a $\alpha = 10$ ó $\alpha = 100$, creciendo este con α a partir de ese punto.

Ridge Regression	Horizonte 1			Horizonte 2		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	0,574	0,570	0,566	1,201	1,183	1,178
Día laborable	0,602	0,583	0,595	1,232	1,165	1,215
Semana previa	0,296	0,296	0,296	0,415	0,414	0,413
Mixta 7 días	0,299	0,299	0,298	0,419	0,419	0,417
Mixta 1 día	0,313	0,313	0,313	0,441	0,441	0,440

Tabla 4.3: MAE del primer y segundo horizonte sobre Ridge Regression (GWh)

Ridge Regression	Horizonte 3			Horizonte 4		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	1,821	1,752	1,773	2,356	2,208	2,268
Día laborable	1,817	1,679	1,786	2,308	2,078	2,252
Semana previa	0,504	0,503	0,500	0,576	0,575	0,571
Mixta 7 días	0,508	0,507	0,504	0,576	0,575	0,570
Mixta 1 día	0,524	0,523	0,522	0,576	0,576	0,575

Tabla 4.4: MAE del tercer y cuarto horizonte sobre Ridge Regression (GWh)

Una vez tomados los valores de α óptimos para cada uno de los casos, los scores obtenidos sobre cada uno de ellos son los que se muestran en las tablas 4.3, 4.4 y 4.5. Cada una de ellas muestra el error absoluto que se espera que tenga la predicción de la demanda de energía eléctrica para cada uno de los horizontes.

Se puede observar cómo, en todos los casos, las tres normalizaciones que hacen uso de proporción entre valores (las de semana previa y las mixtas tanto por 7 días como por 1 día) consiguen los mejores resultados para *Ridge Regression*. De entre estas tres normalizaciones, los resultados para

Ridge Regression	Horizonte 5			Horizonte 6		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	2,733	2,518	2,616	2,949	2,666	2,812
Día laborable	2,643	2,336	2,571	2,818	2,441	2,728
Semana previa	0,630	0,629	0,624	0,672	0,671	0,664
Mixta 7 días	0,625	0,625	0,618	0,666	0,665	0,657
Mixta 1 día	0,613	0,613	0,611	0,645	0,644	0,643

Tabla 4.5: MAE del quinto y sexto horizonte sobre Ridge Regression (GWh)

las primeras horas son mejores con la normalización por semana previa, aproximadamente hasta la cuarta hora. Para la quinta y sexta hora adelante, los mejores resultados los consigue, en este caso, la normalización mixta con un día atrás. Sin embargo, también cabe destacar cómo, para el caso de *Ridge Regression*, sorprendentemente, el proceso de normalización por aproximación a día laborable obtiene la peor predicción para el primer horizonte, aunque a partir del segundo ya supera al estimador sin normalización.

Por otra parte, en cuanto a la adición de los datos sobre temperatura, como cabría esperar, mejora en todos los casos los resultados de las previsiones sin hacer uso de dichos datos. Nótese que, sin embargo, para las previsiones con los datos sin normalizar y con normalización por aproximación a día laborable, la temperatura centrada no realiza una mejora en los resultados a los obtenidos con la temperatura sin tratar, para los últimos horizontes. Para las otras tres normalizaciones sí que se aprecia una mejora.

Una vez obtenidos los resultados, se comparan las series de errores para determinar si hay suficiente evidencia estadística para poder afirmar que un modelo es mejor que otro. Para ello, se hace uso del *Test rank-sum de Wilcoxon* [24]. Sin embargo, realizar una comparación de todos los modelos sería un proceso innecesariamente largo. Con ésto en mente se sigue la siguiente estrategia: para cada horizonte, se comparan los dos modelos con menor *MAE*. En el caso de los modelos de *Ridge Regression*, para los cuatro primeros horizontes, no se aprecia suficiente evidencia estadística a nivel 0,01 para seleccionar entre la normalización por semana previa y la mixta a 7 días, ambas con temperatura centrada; sin embargo, para el quinto y sexto horizonte sí la hay para elegir la mixta a 1 día sobre la mixta a 7 días, ambas con temperatura centrada. Para más detalle se puede consultar la tabla C.1 adjunta en el anexo C.

4.4.2. Experimentos sobre Perceptrones Multicapa

El modelo de *Multilayer Perceptron* es un poco más complejo que el de *Ridge Regression*; sin embargo, a la hora de hiperparametrizar solo se cuenta con dos valores a probar: la configuración de capas y λ . Para su hiperparametrización se han probado dos configuraciones distintas: una con una sola capa oculta de 100 neuronas, y otra con dos capas ocultas, la primera de 100 neuronas y la

segunda de 20. Por su parte, para λ se tomó una regilla con valores de la forma 10^k para $k = -8, \dots, 3$. Los CV MAE de la hiperparametrización de cada uno de los casos pueden observarse en las figuras B.4, B.5 y B.6 adjuntas en el anexo B. En todos los casos el comportamiento es similar al observado en la hiperparametrización de *Ridge Regression*: se encuentran unos valores del CV MAE estables hasta en torno a $\lambda = 10$, creciendo este con λ a partir de ese punto. Dependiendo de la normalización y los datos, sí se puede apreciar una diferencia en la configuración de capas que minimiza el CV MAE.

MLP	Horizonte 1			Horizonte 2		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	0,439	0,445	0,477	0,735	0,716	0,786
Día laborable	0,475	0,434	0,491	0,762	0,674	0,766
Semana previa	0,298	0,302	0,304	0,414	0,407	0,412
Mixta 7 días	0,301	0,304	0,304	0,419	0,413	0,413
Mixta 1 día	0,309	0,310	0,318	0,428	0,427	0,431

Tabla 4.6: MAE del primer y segundo horizonte sobre MLP (GWh)

MLP	Horizonte 3			Horizonte 4		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	0,957	0,925	1,003	1,151	1,120	1,187
Día laborable	0,960	0,814	0,922	1,082	0,916	1,052
Semana previa	0,504	0,487	0,493	0,576	0,550	0,560
Mixta 7 días	0,508	0,495	0,497	0,576	0,555	0,561
Mixta 1 día	0,508	0,506	0,505	0,564	0,560	0,557

Tabla 4.7: MAE del tercer y cuarto horizonte sobre MLP (GWh)

MLP	Horizonte 5			Horizonte 6		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	1,306	1,275	1,337	1,393	1,379	1,444
Día laborable	1,172	1,017	1,163	1,231	1,078	1,262
Semana previa	0,631	0,597	0,611	0,675	0,636	0,654
Mixta 7 días	0,627	0,601	0,608	0,668	0,635	0,648
Mixta 1 día	0,602	0,599	0,597	0,632	0,629	0,622

Tabla 4.8: MAE del quinto y sexto horizonte sobre MLP (GWh)

Una vez tomada la mejor configuración de capas y el λ óptimo para cada uno de los casos, los scores obtenidos sobre cada uno de ellos son los que se muestran en las tablas 4.6, 4.7 y 4.8. Cada una de ellas muestra el error absoluto que se espera que tenga la predicción de la demanda de energía eléctrica para cada uno de los horizontes.

Se puede observar cómo el comportamiento de los distintos modelos de *Multilayer Perceptron* es similar al de los modelos de *Ridge Regression*: las tres normalizaciones que hacen uso de proporción entre valores consiguen los mejores resultados. Durante las primeras horas los mejores resultados los obtiene la normalización por semana previa y las últimas, a partir del cuarto horizonte, los obtiene la normalización mixta con un día atrás. También cabe destacar que, para *Multilayer Perceptron*, la

normalización por aproximación a día laborable mejora al estimador sin normalización desde el primer horizonte.

En cuanto a la adición de la temperatura centrada, sorprende que para el primer horizonte no mejore los resultados, aunque con el paso de éstos sí que se aprecia una clara mejora para todos los procesos de normalización. Sin embargo, en ningún caso mejora los resultados para el modelo sin normalizar.

Una vez obtenidos los resultados, se comparan las series de errores para determinar si hay suficiente evidencia estadística para poder afirmar que un modelo es mejor que otro. Se sigue el mismo procedimiento explicado para *Ridge Regression*. En el caso de los modelos de *Multilayer Perceptron*, hasta el último horizonte no se aprecia suficiente evidencia estadística a nivel 0,01 para seleccionar un modelo sobre otro. El primer horizonte se compararon los modelos de normalización por semana previa y mixta a 7 días sin temperatura; los siguientes tres horizontes con temperatura; y el quinto la mixta a 1 día con temperatura centrada y semana previa con temperatura sin centrar. Finalmente, en el sexto horizonte sí se encuentra suficiente evidencia estadística para seleccionar la mixta a 1 día con temperatura centrada sobre la mixta a 7 días con temperatura sin centrar. Para más detalle se puede consultar la tabla C.2 adjunta en el anexo C.

4.4.3. Experimentos sobre SVR

La hiperparametrización de los modelos de *SVR* es la más compleja de las tres realizadas, al contar con tres hiperparámetros: C , γ y ϵ . Para los tres, se usaron distintas regillas:

- Para C con valores de la forma 10^k para $k = -3, \dots, 3$.
- Para γ con valores de la forma $2^k/d$ para $k = -2, \dots, 6$ y donde d denota la cantidad de parámetros de entrada.
- Para ϵ con valores de la forma 2^k para $k = -6, \dots, -1$.

Los CV MAE pueden observarse en las figuras B.7, B.8 y B.9 en función de C , en las figuras B.10, B.11 y B.12 en función de γ y en las figuras B.13, B.14 y B.15 en función de ϵ , todas ellas adjuntas en el anexo B.

SVR	Horizonte 1			Horizonte 2		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	0,432	0,416	0,426	0,731	0,704	0,725
Día laborable	0,434	0,415	0,445	0,752	0,710	0,727
Semana previa	0,293	0,290	0,292	0,410	0,400	0,403
Mixta 7 días	0,299	0,296	0,298	0,416	0,408	0,411
Mixta 1 día	0,301	0,297	0,300	0,414	0,407	0,411

Tabla 4.9: MAE del primer y segundo horizonte sobre SVR (GWh)

En este caso, al hacer uso de la clase *MultiOutputRegressor* de *sklearn*, se han tomado como los mejores hiperparámetros aquellos valores que optimizan el promedio de los CV MAE de los seis

SVR	Horizonte 3			Horizonte 4		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	0,961	0,928	0,963	1,143	1,116	1,155
Día laborable	0,967	0,894	0,889	1,098	0,995	1,007
Semana previa	0,499	0,482	0,488	0,571	0,547	0,556
Mixta 7 días	0,506	0,490	0,495	0,573	0,550	0,556
Mixta 1 día	0,491	0,483	0,488	0,545	0,536	0,541

Tabla 4.10: MAE del tercer y cuarto horizonte sobre SVR (GWh)

SVR	Horizonte 5			Horizonte 6		
Normalización	ST	CT	CTC	ST	CT	CTC
Sin normalización	1,303	1,279	1,312	1,431	1,404	1,436
Día laborable	1,203	1,085	1,095	1,270	1,139	1,152
Semana previa	0,628	0,597	0,608	0,672	0,638	0,648
Mixta 7 días	0,625	0,596	0,604	0,667	0,633	0,643
Mixta 1 día	0,585	0,576	0,580	0,616	0,605	0,609

Tabla 4.11: MAE del quinto y sexto horizonte sobre SVR (GWh)

horizontes. Los score obtenidos sobre cada uno de ellos son los que se muestran en las tablas 4.9, 4.10 y 4.11. Cada una de ellas muestra el error absoluto que se espera que tenga la predicción de la demanda de energía eléctrica para cada uno de los horizontes.

Los resultados reflejan los mismos comportamientos que en los casos de *Ridge Regression* y *Multilayer Perceptron* en cuanto a cómo afectan las normalizaciones y la adición de los datos de temperatura sin centrar y centrada. Sin embargo, en el caso de SVR, sí que es verdad que el centrado de la temperatura no mejora en ningún caso a la temperatura sin centrar.

Una vez obtenidos los resultados, al igual que para *Ridge Regression* y *Multilayer Perceptron*, se comparan las series de errores para determinar si hay suficiente evidencia estadística para poder afirmar que un modelo es mejor que otro. En el caso de los modelos de SVR, en los primeros tres horizontes no se aprecia suficiente evidencia estadística a nivel 0,01 para seleccionar la normalización por semana previa sobre las distintas mixtas, en todos los casos con temperatura sin centrar; sin embargo, para los tres últimos horizontes sí que hay suficiente evidencia para seleccionar la normalización mixta a 1 día con temperatura sin centrar. Para más detalle se puede consultar la tabla C.3 adjunta en el anexo C.

Finalmente, sí que cabe destacar que, aunque cómo evolucionan los resultados a lo largo de los horizontes y en función de normalizaciones y temperatura es equivalente para los tres modelos de *Machine Learning* estudiados, se aprecia una mejora primero al pasar de *Ridge Regression* a *Multilayer Perceptron*, llegando finalmente a los mejores resultados con SVR. Para más detalle se puede consultar la tabla C.4 adjunta en el anexo C. Sobre estas comparaciones se discutirá en mayor profundidad en el capítulo 5 de conclusiones.

CONCLUSIONES Y TRABAJO FUTURO

Finalmente, una vez obtenidos los resultados prácticos del trabajo y discutido sobre ellos, es momento de repasar lo expuesto a lo largo de todo el trabajo. En primer lugar, en la sección 5.1 se revisarán los resultados obtenidos. Para terminar, en la sección 5.2, se propondrán posibles ampliaciones futuras del trabajo.

5.1. Conclusiones

A modo de conclusión, en este trabajo se ha expuesto la importancia que tiene en un sistema eléctrico encontrar el equilibrio entre la demanda energética y la generación, y cómo la previsión de la demanda juega un papel fundamental en el proceso. Más concretamente, en el capítulo 2, se ha explicado el concepto de sistema eléctrico, los elementos y agentes que lo conforman, y las características concretas del sistema eléctrico español. Adicionalmente, también se propusieron distintos procesos de normalización posibles para los datos de demanda de energía eléctrica que puedan mejorar los resultados en las posteriores estimaciones, basados en características concretas de su serie temporal.

Posteriormente, se discutió cómo los distintos modelos de *machine learning* pueden aportar un mecanismo para la previsión de la demanda de energía eléctrica. A lo largo del capítulo 3 se explicó el comportamiento teórico de los estimadores de *Ridge Regression*, *Multilayer Perceptron* y *Support Vector Regressor*, modelos que posteriormente fueron utilizados en los experimentos prácticos del trabajo.

Finalmente, en el capítulo 4 se explicó la realización de los experimentos sobre los datos reales de demanda energética en España. Se desarrolló primero la selección de *Python* como lenguaje de programación principal, explicando los distintos paquetes de los que dispone como principal motivo de su elección. También se discutió la obtención de los datos y su posterior procesado, tanto para la serie temporal de demanda de energía eléctrica como para la de temperatura.

Después se realizó un análisis del tratamiento de los datos. En ese sentido, se discutió en qué medida se esperaba que dichos procesos de normalización de los datos de la demanda energética

y centrado de la temperatura afectarán a la posterior estimación. Para terminar, se realizaron los experimentos propiamente dichos sobre los modelos de *machine learning* expuestos con anterioridad. También se discutieron los resultados y cómo realmente afectaron las normalizaciones y la adición de los datos de temperatura. Sobre los resultados de los diversos modelos, se realizaron *Tests de Wilcoxon* para determinar si había suficiente evidencia estadística para seleccionar un modelo sobre otro. Más concretamente, se aplicó su versión en la que se compara si una de las distribuciones de errores especificada es mayor que la otra (parámetro *alternative* = 'less' en la función *ranksum()* del paquete de *Python scipy*).

Como conclusión final, cabe destacar que los modelos de *SVR* obtienen los mejores resultados, sobre los modelos de *Multilayer Perceptron* y *Ridge Regression*. En concreto, estos modelos de *SVR* se ven positivamente afectados por las normalizaciones por semana previa y mixtas, y la adición de los datos de temperatura sin centrar. Como se puede apreciar en la tabla C.4 adjunta en el anexo C, hay suficiente evidencia estadística para realizar esta afirmación, a nivel 0,01, para el primer horizonte, y los tres últimos.

Además, es posible que la normalización por aproximación a día laborable no obtenga resultados especialmente buenos debido a que los coeficientes empleados en dicho proceso son estáticos y hacen uso de datos alejados en el tiempo del momento sobre el que se desea realizar la estimación (los coeficientes son valores calculados sobre los datos de 2021 y 2022, mientras que el conjunto de *test* son las observaciones de 2023). Sin embargo, la normalización por semana previa y las mixtas siguen la evolución de la serie de manera más cercana.

5.2. Trabajo futuro

En un futuro, se podrían estudiar nuevos procesos de normalización y nuevas variables a añadir en el proceso de estimación de la demanda de energía eléctrica. También se podría indagar en mayor profundidad en los motivos por los que la normalización por aproximación a día laborable o el centrado de temperatura no realizan mejoras sustanciales en los resultados.

Por otro lado, también se podría estudiar el uso de otros modelos de *machine learning* más potentes, o que se adecúen mejor al problema de estimación de series temporales. Adicionalmente, tal y como se explicó en el capítulo 4, Red Eléctrica de España realiza sus estimaciones con modelos que se van actualizando a lo largo del tiempo. En ese sentido, otro campo de ampliación de este trabajo podría ser añadir procesos de actualización de los estimadores utilizados.

BIBLIOGRAFÍA

- [1] Red Eléctrica de España, “La operación del sistema eléctrico para dummies,” in *Centro Libros PAPP*, SLU Barcelona, 2013.
- [2] “Boe nº312, de 29 de diciembre de 1984.” https://www.boe.es/diario_boe/txt.php?id=BOE-A-1984-28282. [Online; accessed 6-October-2023].
- [3] Red Eléctrica de España, “Red eléctrica en 2 minutos.” <https://www.ree.es/es/conocenos/ree-en-2-minutos>. [Online; accessed 7-December-2023].
- [4] Red Eléctrica de España, “Actividades de negocio.” <https://www.ree.es/es/conocenos/actividades-de-negocio>. [Online; accessed 7-December-2023].
- [5] Red Eléctrica de España, “Operación del sistema eléctrico.” <https://www.ree.es/es/actividades/operacion-del-sistema-electrico>. [Online; accessed 7-December-2023].
- [6] T. Hastie, R. Tibshirani, and J. Friedman, *Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, New York: Springer, second edition ed., 2009.
- [7] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, 2006.
- [8] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
- [9] D. E. Rumelhart, J. L. McClelland, and P. R. Group, *Parallel Distributed Processing, Volume 1: Explorations in the Microstructure of Cognition: Foundations*. The MIT Press, 07 1986.
- [10] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2017.
- [11] V. Vapnik, *The Nature of Statistical Learning Theory*. Information Science and Statistics, Springer New York, 1999.
- [12] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 06 2018.
- [13] J. Lopez and J. R. Dorronsoro, “Simple proof of convergence of the smo algorithm for different svm variants,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 7, pp. 1142–1147, 2012.
- [14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [15] requests, “Requests: HTTP for Humans.” <https://requests.readthedocs.io/en/latest/>. [Online; accessed 6-August-2023].
- [16] meteostat, “Python Library | Meteostat Developers.” <https://dev.meteostat.net/python/>. [Online; accessed 12-September-2023].

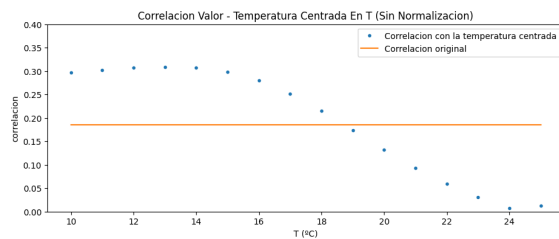
- [17] pandas, “pandas - Python Data Analysis Library.” <https://pandas.pydata.org/>. [Online; accessed 6-August-2023].
- [18] numpy, “NumPy -.” <https://numpy.org/>. [Online; accessed 6-August-2023].
- [19] matplotlib, “Matplotlib – Visualization with Python.” <https://matplotlib.org/>. [Online; accessed 5-November-2023].
- [20] hvplot, “hvPlot – hvPlot documentation.” <https://hvplot.holoviz.org/>. [Online; accessed 5-November-2023].
- [21] scipy, “scipy.stats.ranksums – SciPy Manual.” <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ranksums.html>. [Online; accessed 28-March-2024].
- [22] calendarioslaborales.com, “Calendarios laborales de españa por provincias.” <https://www.calendarioslaborales.com/>. [Online; accessed 15-October-2023].
- [23] Wikipedia, “Anexo: Provincias y ciudades autónomas de españa.” https://es.wikipedia.org/wiki/Anexo:Provincias_y_ciudades_aut%C3%B3nomas_de_Espa%C3%B1a. [Online; accessed 15-October-2023].
- [24] F. Wilcoxon, “Individual comparisons by ranking methods,” *BIOMETRICS BULLETIN*, vol. 1, no. 6, pp. 80–83, 1945.

APÉNDICES

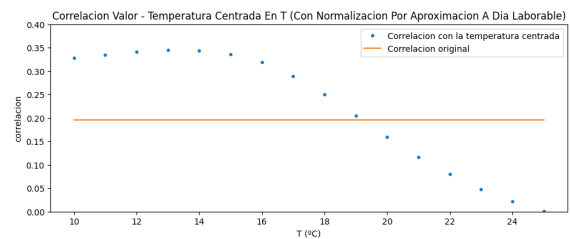
RESULTADOS DEL PROCESADO DE LOS DATOS

DATOS

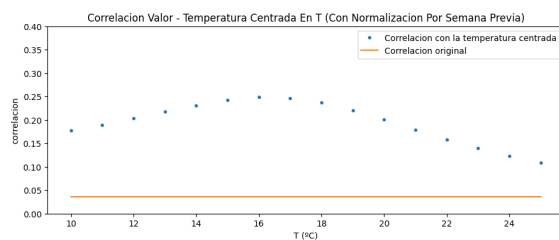
En este anexo se presentarán las distintas figuras resultado tanto de los distintos procesos de normalización de los datos de demanda de energía eléctrica como del centrado de los datos de temperatura. Por un lado, en cada una de las imágenes que conforman la figura A.1 los puntos muestran la correlación de los datos con la temperatura centrada en ese punto, mientras que la línea indica la correlación con la temperatura sin tratar.



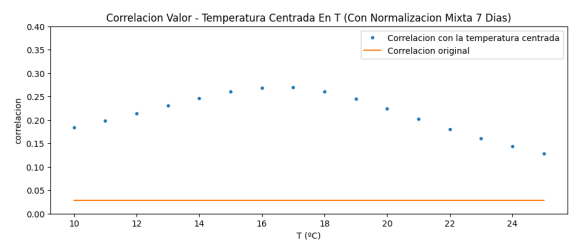
(a) Sin normalización



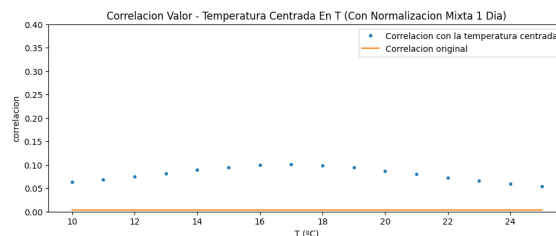
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa



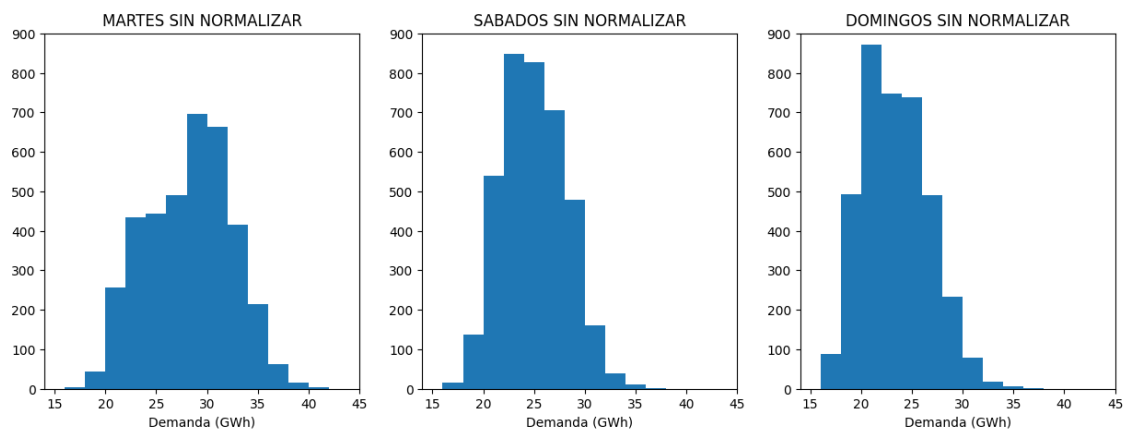
(d) Con normalización mixta 7 días



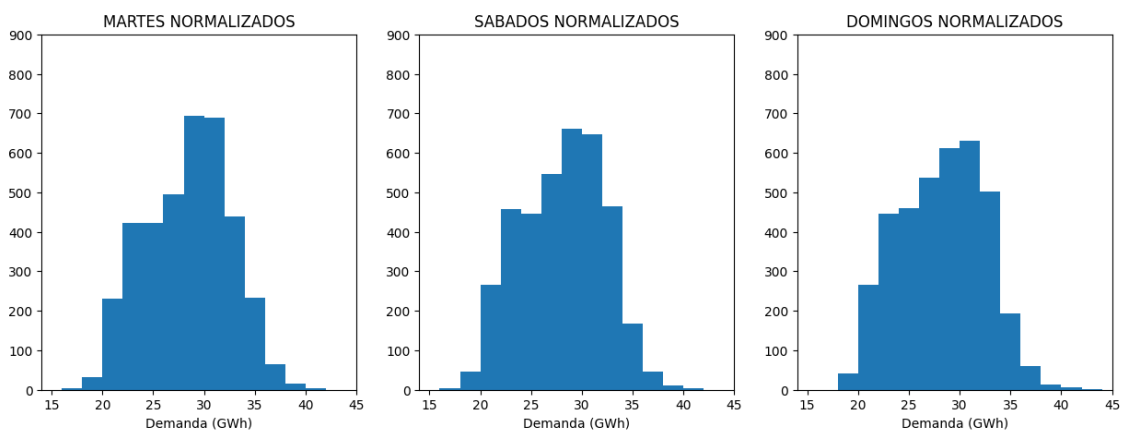
(e) Con normalización mixta 1 día

Figura A.1: Búsqueda de la insensibilidad térmica

Por otra parte, las figuras A.2, A.3 y A.4 muestran los histogramas realizados para el análisis de los resultados de los procesos de normalización de la demanda energética.

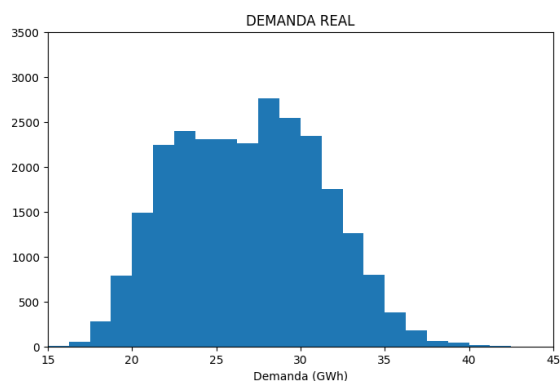


(a) Sin normalización

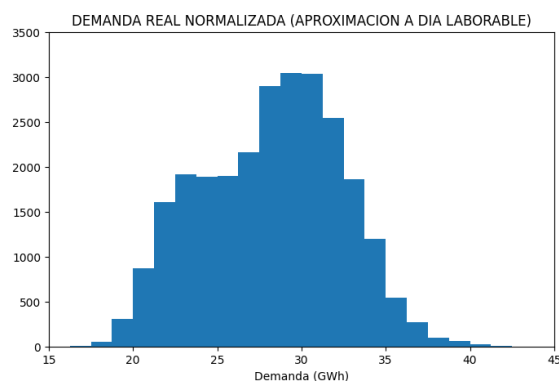


(b) Con normalización por aproximación a día laborable

Figura A.2: Histogramas específicos de los datos de demanda energética sin normalizar y normalizados por aproximación a día laborable los martes, sábados y domingos

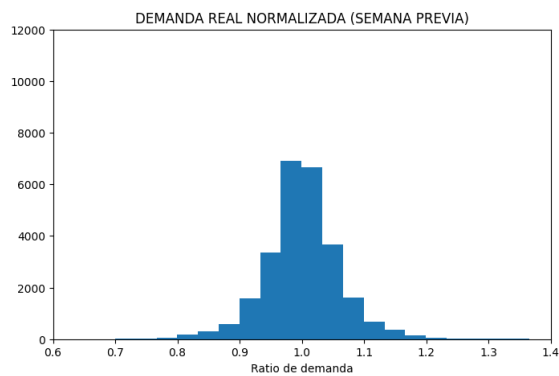


(a) Sin normalización

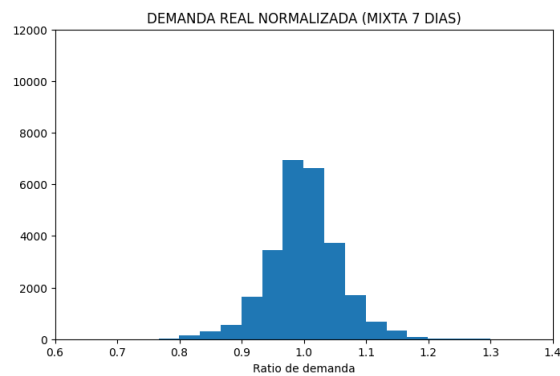


(b) Con normalización por aproximación a día laborable

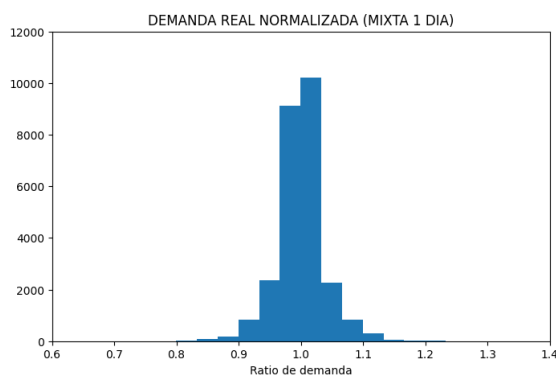
Figura A.3: Histogramas generales de los datos de demanda energética sin normalizar y con normalización por aproximación a día laborable



(a) Con normalización por semana previa



(b) Con normalización mixta 7 días



(c) Con normalización mixta 1 día

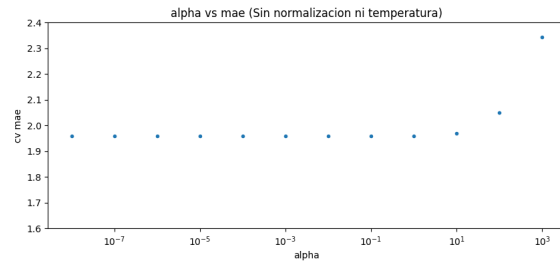
Figura A.4: Histogramas generales de los datos de demanda energética normalizados por semana previa y por normalización mixta

HIPERPARAMETRIZACIONES

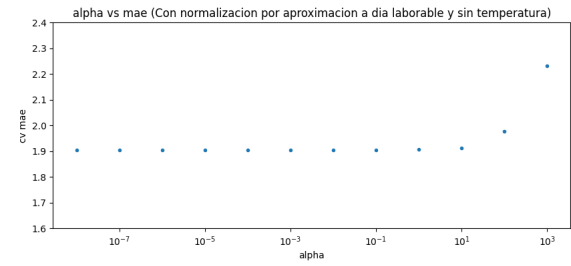
En este anexo se presentarán las distintas figuras resultado de realizar el proceso de hiperparametrización sobre los distintos modelos, y que son referenciadas en las explicaciones sobre los experimentos realizados. En primer lugar, en las figuras B.1, B.2 y B.3 se pueden observar los resultados de la hiperparametrización del parámetro α de *Ridge Regression* sobre los modelos que no cuentan con datos de temperatura, los que cuentan con los datos de temperatura sin tratar y por último los que cuentan con los datos de temperatura centrada.

Por su parte, en las figuras B.4, B.5 y B.6 se pueden observar los resultados de la hiperparametrización del parámetro λ de *Multilayer Perceptron* sobre los modelos que no cuentan con datos de temperatura, los que cuentan con los datos de temperatura sin tratar y por último los que cuentan con los datos de temperatura centrada. Se distinguen a su vez entre los resultados de las dos configuraciones de capas ocultas probadas.

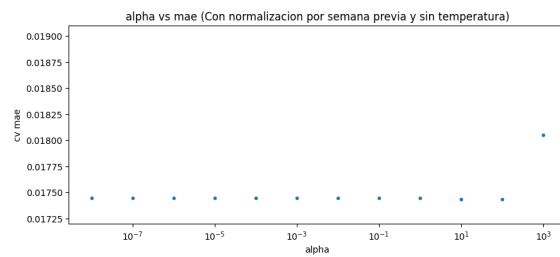
Finalmente, en las figuras B.7, B.8, B.9, B.10, B.11, B.12, B.13, B.14 y B.15 se pueden observar los resultados de la hiperparametrización de los parámetros de *SVR*. Las tres primeras son en función de C , las tres segundas en función de γ y las tres últimas en función de ϵ . Dentro de esos grupos de tres figuras mencionados anteriormente, en cada uno de ellos, la primera figura corresponde a datos sin temperatura, la segunda a datos con temperatura y la tercera y última a datos con temperatura centrada.



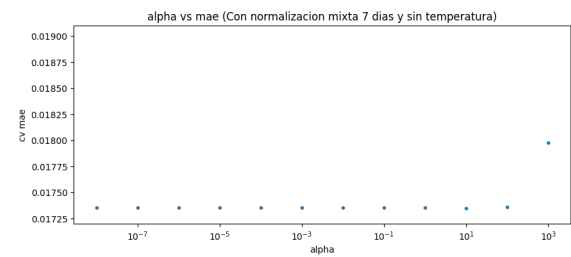
(a) Sin normalización



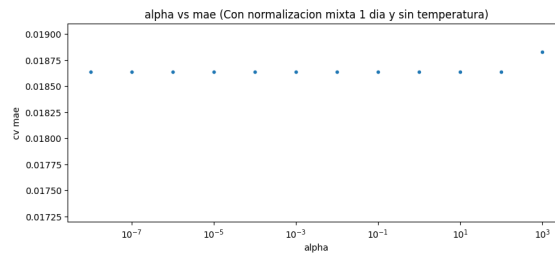
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

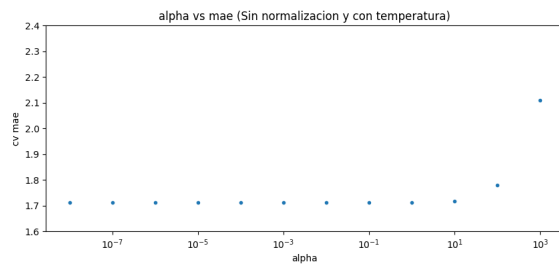


(d) Con normalización mixta 7 días

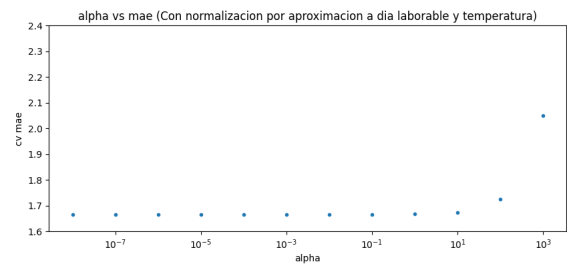


(e) Con normalización mixta 1 día

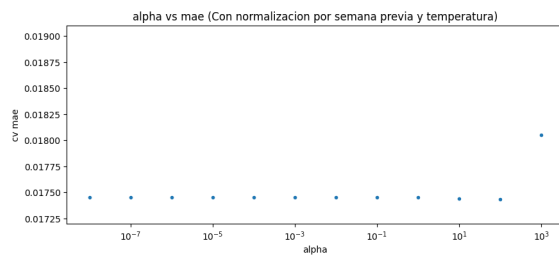
Figura B.1: Hiperparametrización de Ridge Regression sin temperatura



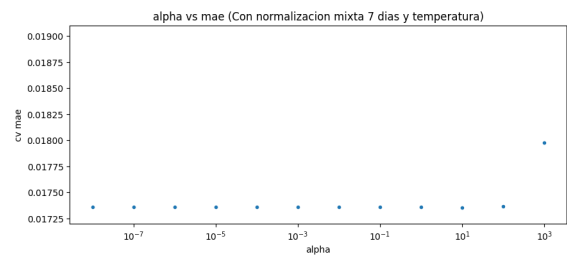
(a) Sin normalización



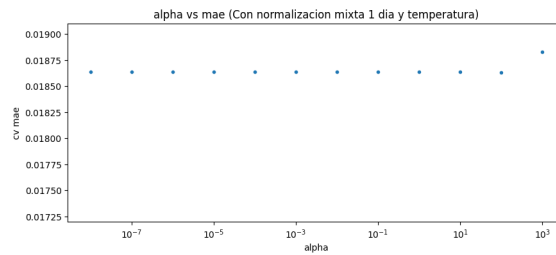
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

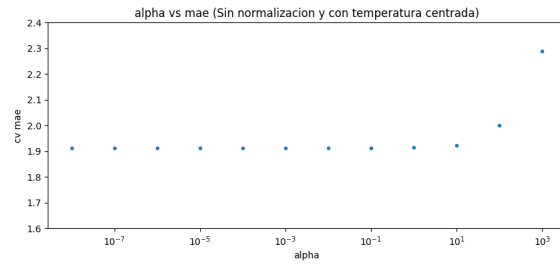


(d) Con normalización mixta 7 días

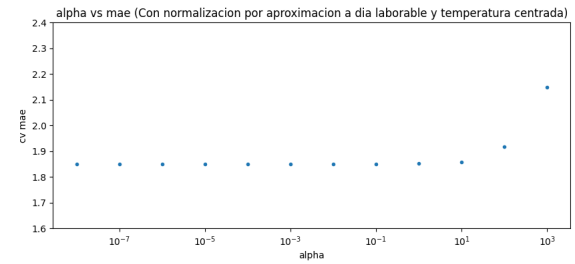


(e) Con normalización mixta 1 día

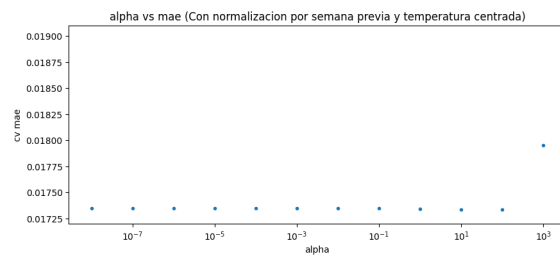
Figura B.2: Hiperparametrización de Ridge Regression con temperatura



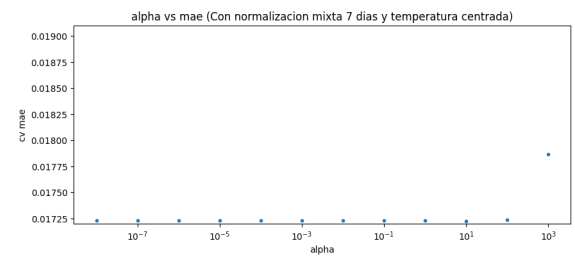
(a) Sin normalización



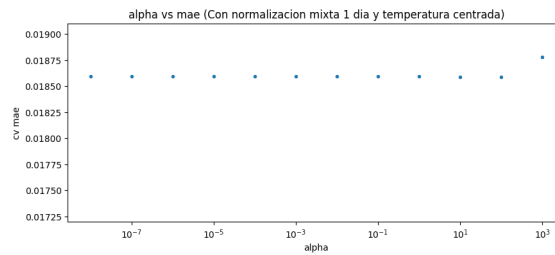
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

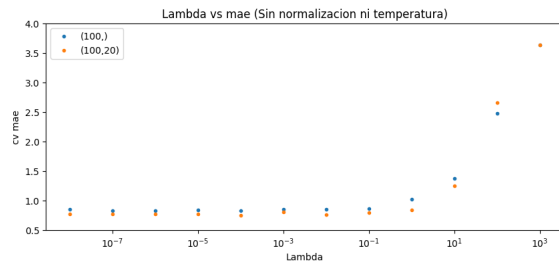


(d) Con normalización mixta 7 días

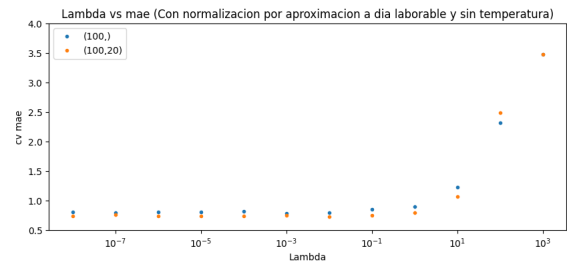


(e) Con normalización mixta 1 día

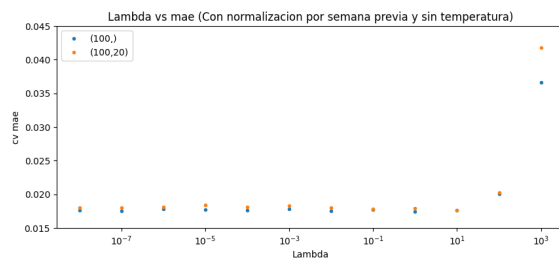
Figura B.3: Hiperparametrización de Ridge Regression con temperatura centrada



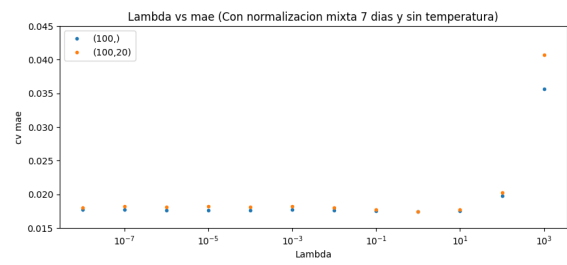
(a) Sin normalización



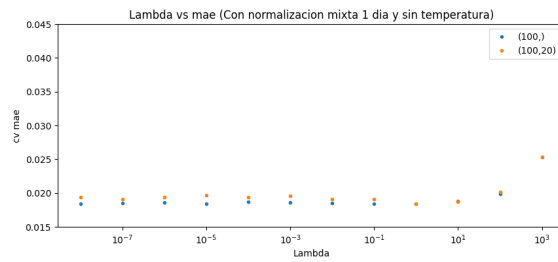
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

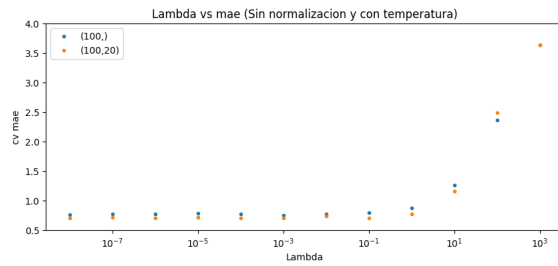


(d) Con normalización mixta 7 días

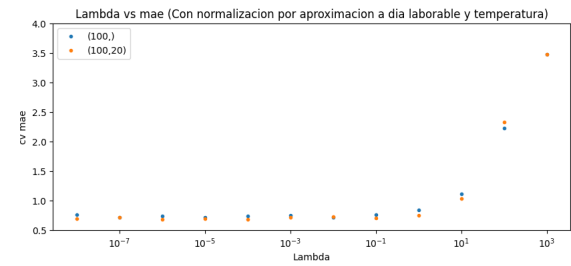


(e) Con normalización mixta 1 día

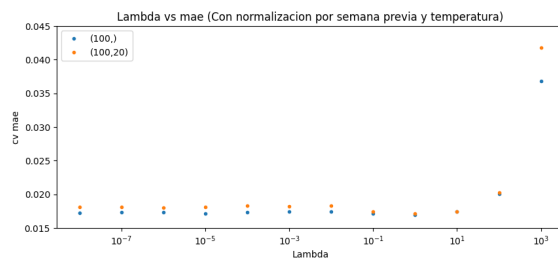
Figura B.4: Hiperparametrización de Multilayer Perceptron sin temperatura



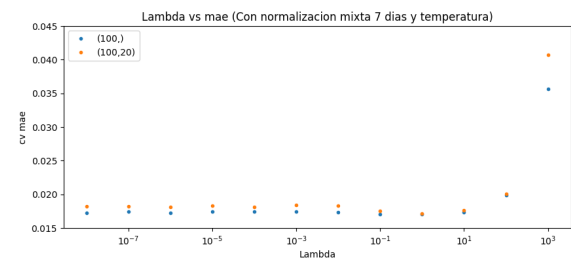
(a) Sin normalización



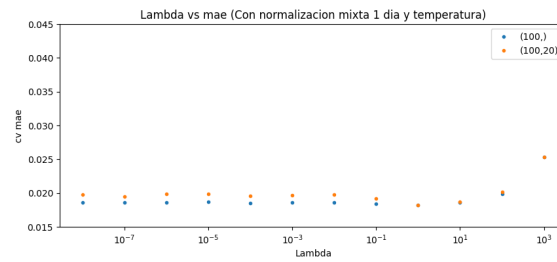
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

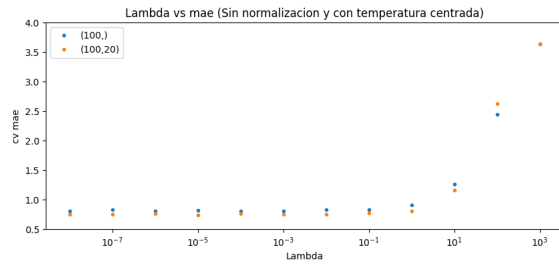


(d) Con normalización mixta 7 días

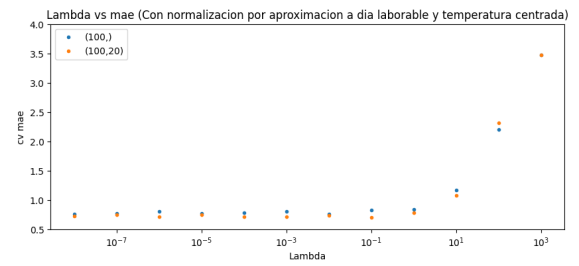


(e) Con normalización mixta 1 día

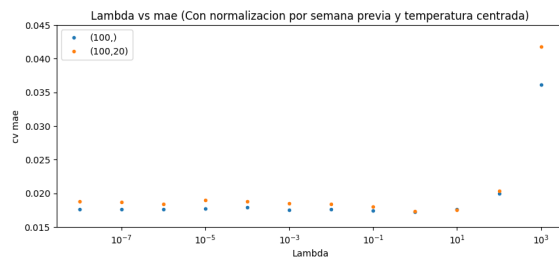
Figura B.5: Hiperparametrización de Multilayer Perceptron con temperatura



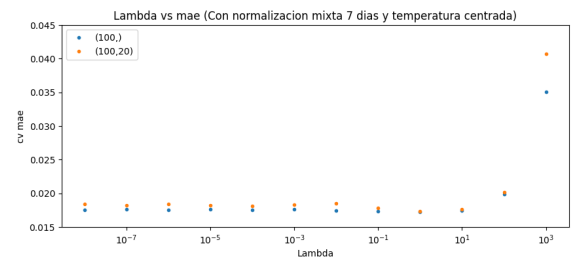
(a) Sin normalización



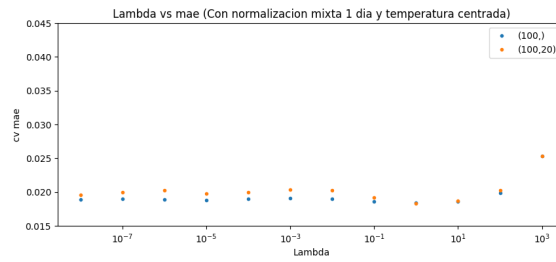
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

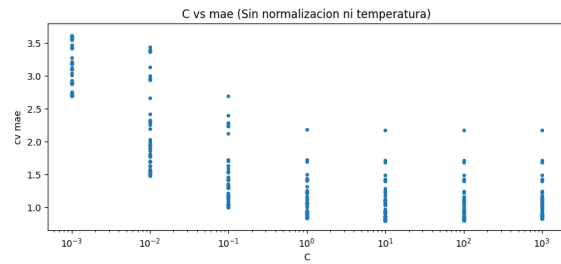


(d) Con normalización mixta 7 días

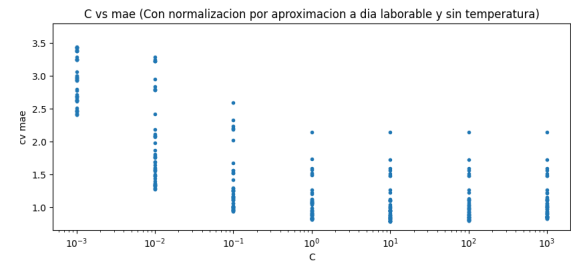


(e) Con normalización mixta 1 día

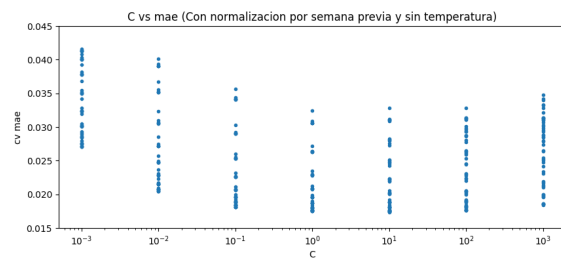
Figura B.6: Hiperparametrización de Multilayer Perceptron con temperatura centrada



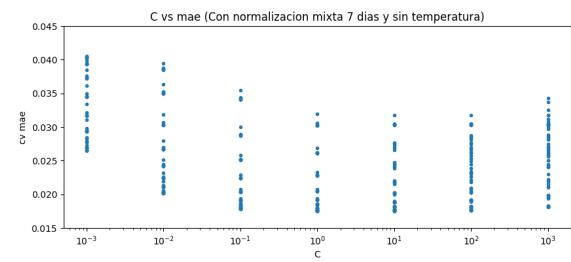
(a) Sin normalización



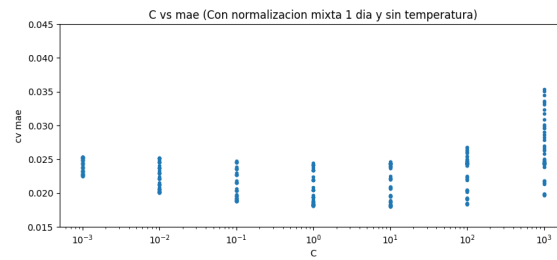
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

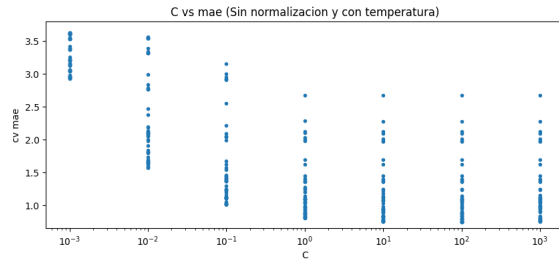


(d) Con normalización mixta 7 días

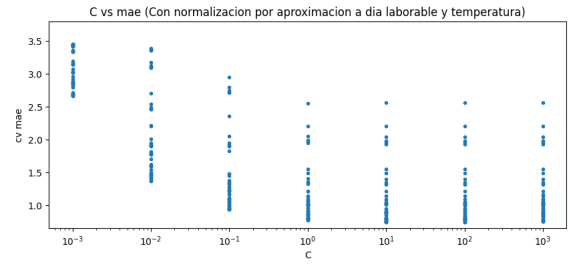


(e) Con normalización mixta 1 día

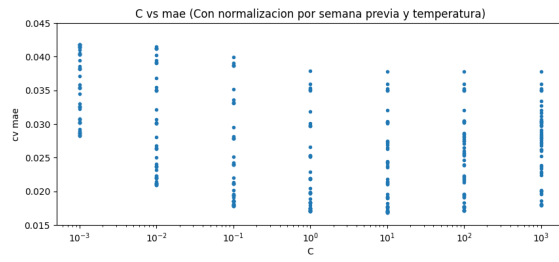
Figura B.7: Hiperparametrización de SVR sin temperatura en función de C



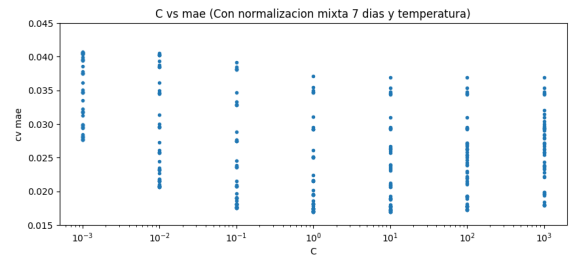
(a) Sin normalización



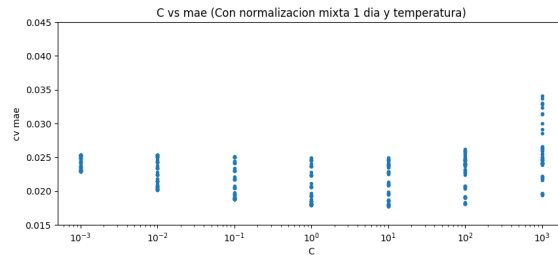
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

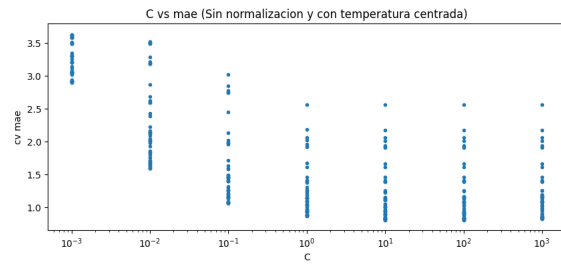


(d) Con normalización mixta 7 días

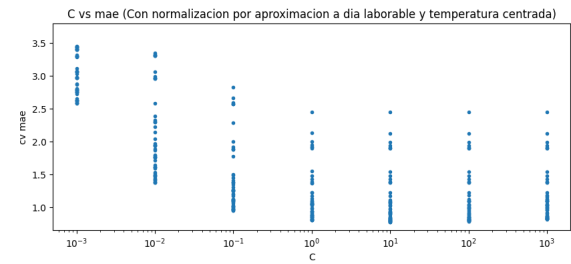


(e) Con normalización mixta 1 día

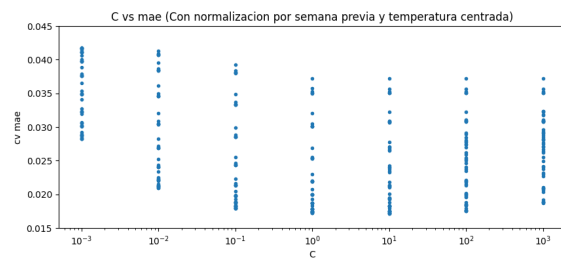
Figura B.8: Hiperparametrización de SVR con temperatura en función de C



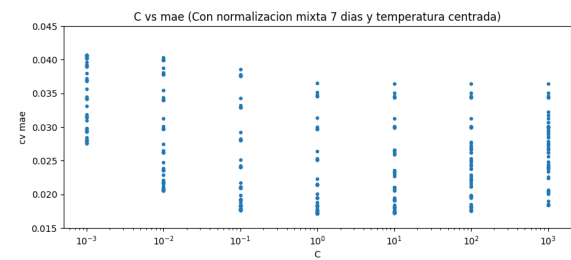
(a) Sin normalización



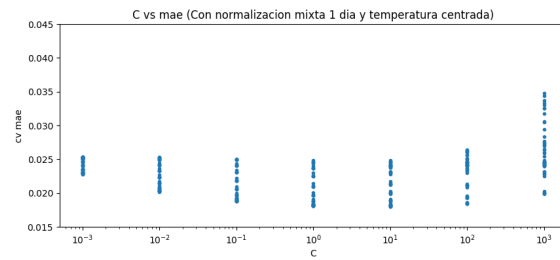
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

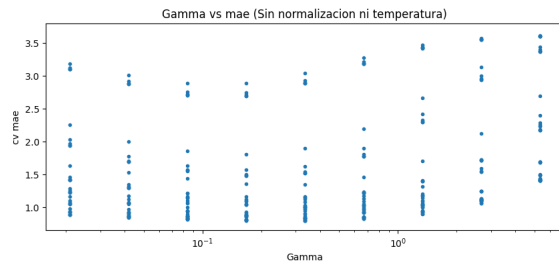


(d) Con normalización mixta 7 días

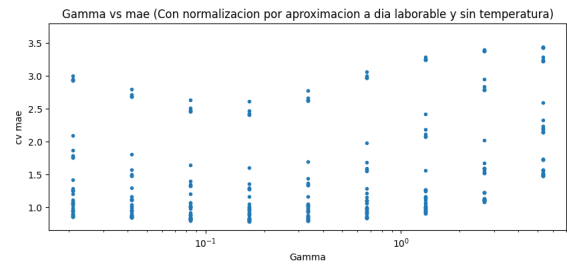


(e) Con normalización mixta 1 día

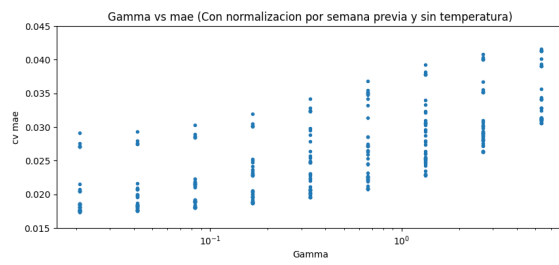
Figura B.9: Hiperparametrización de SVR con temperatura centrada en función de C



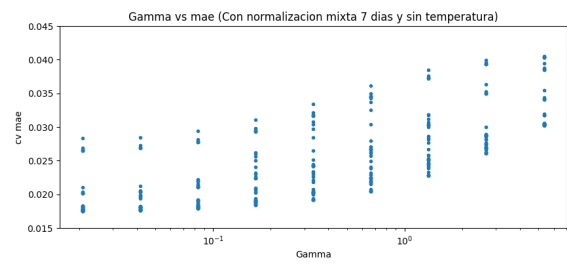
(a) Sin normalización



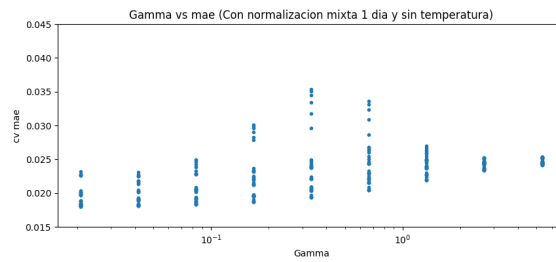
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

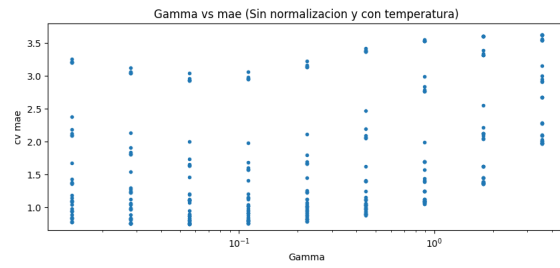


(d) Con normalización mixta 7 días

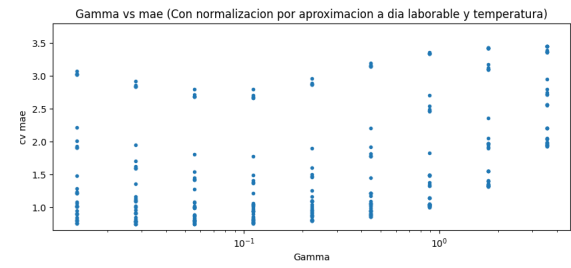


(e) Con normalización mixta 1 día

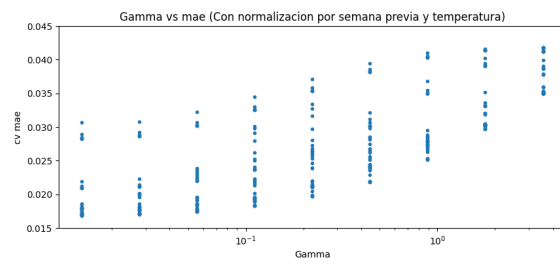
Figura B.10: Hiperparametrización de SVR sin temperatura en función de γ



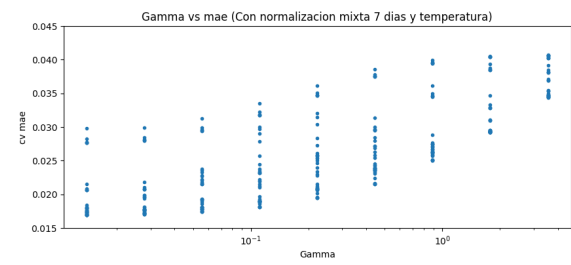
(a) Sin normalización



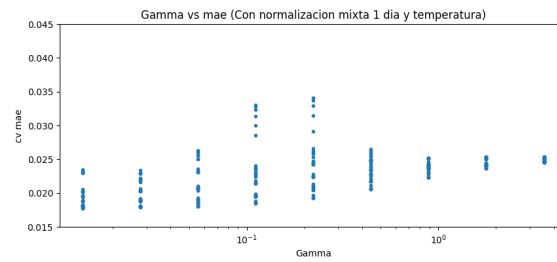
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

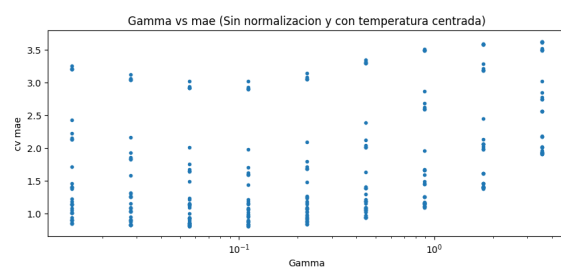


(d) Con normalización mixta 7 días

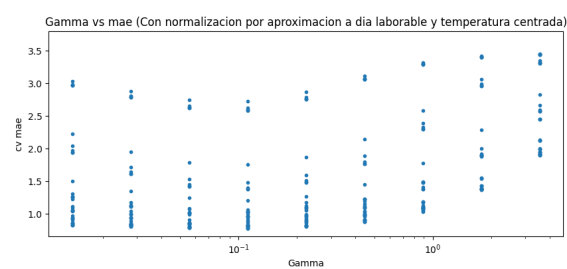


(e) Con normalización mixta 1 día

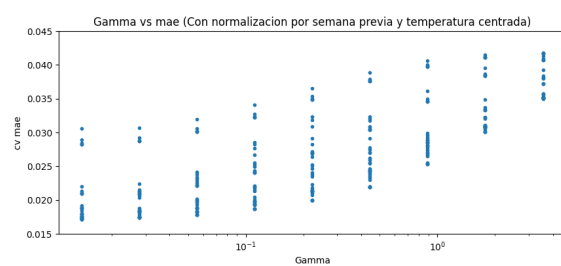
Figura B.11: Hiperparametrización de SVR con temperatura en función de γ



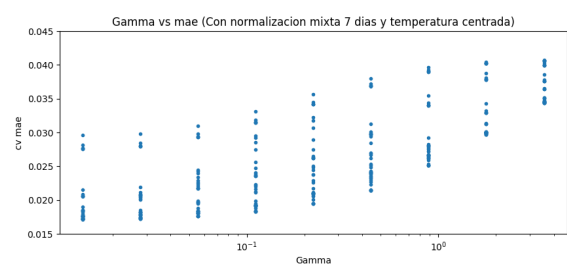
(a) Sin normalización



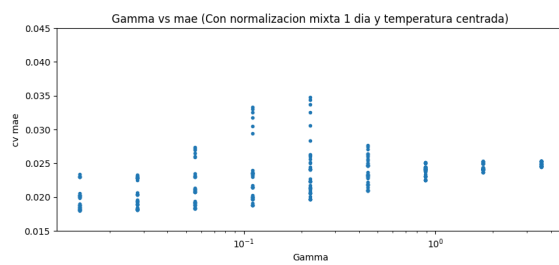
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

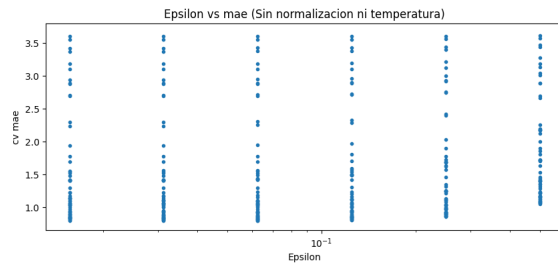


(d) Con normalización mixta 7 días

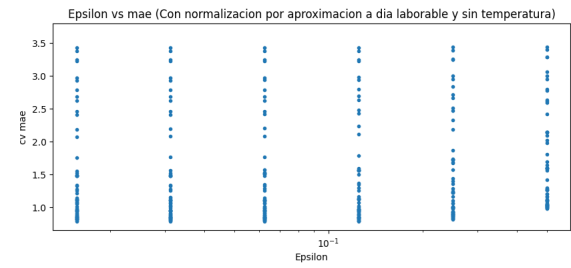


(e) Con normalización mixta 1 día

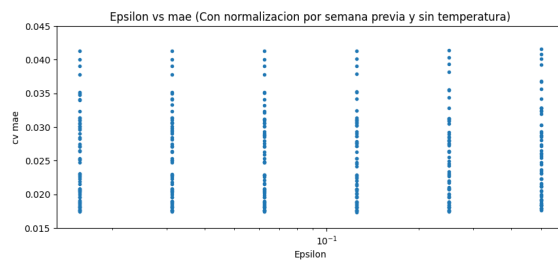
Figura B.12: Hiperparametrización de SVR con temperatura centrada en función de γ



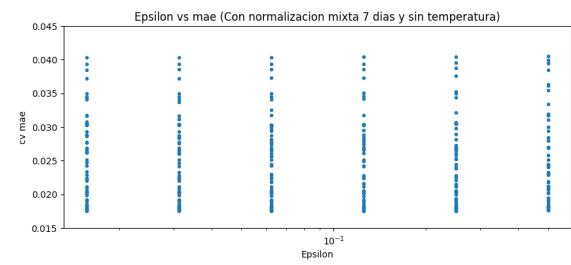
(a) Sin normalización



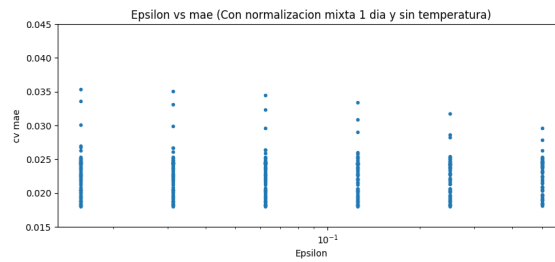
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

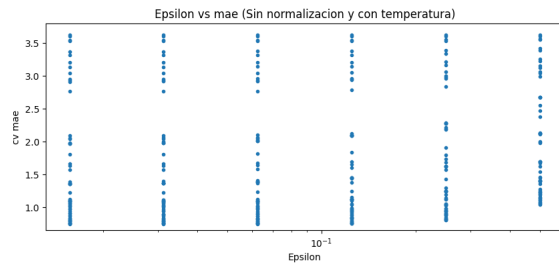


(d) Con normalización mixta 7 días

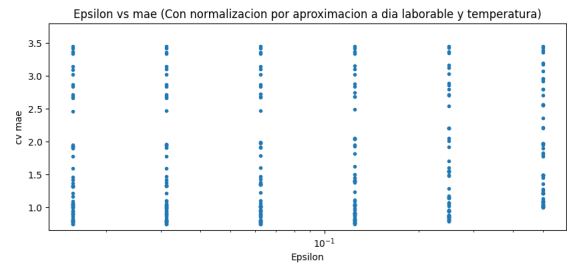


(e) Con normalización mixta 1 día

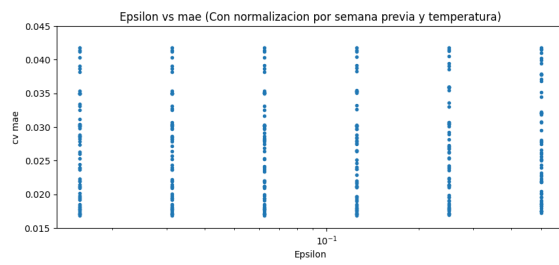
Figura B.13: Hiperparametrización de SVR sin temperatura en función de ϵ



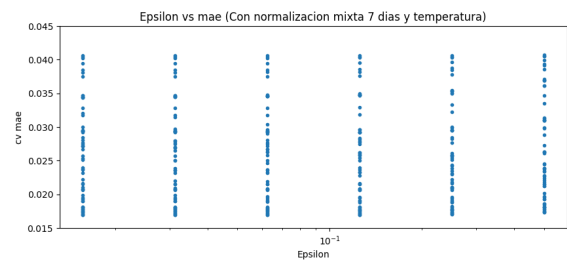
(a) Sin normalización



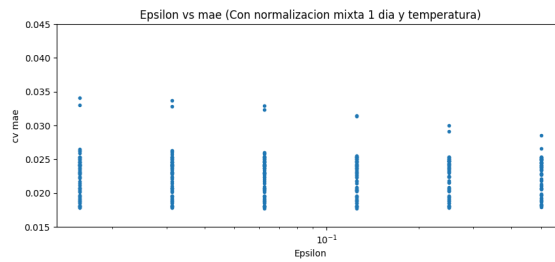
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa

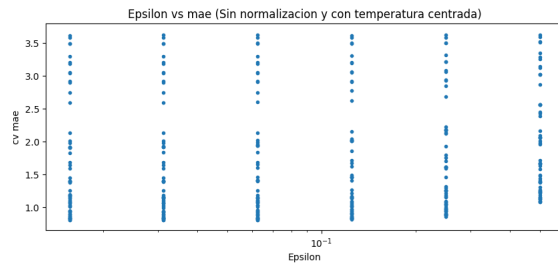


(d) Con normalización mixta 7 días

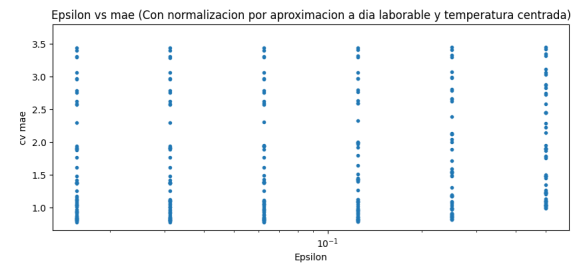


(e) Con normalización mixta 1 día

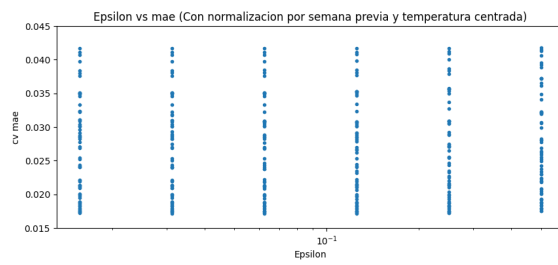
Figura B.14: Hiperparametrización de SVR con temperatura en función de ϵ



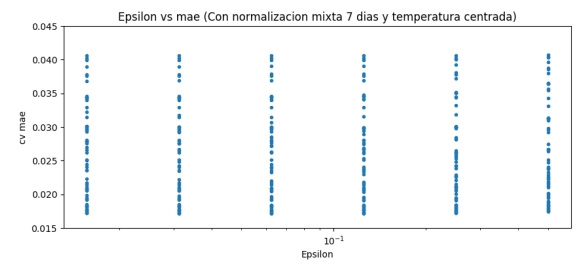
(a) Sin normalización



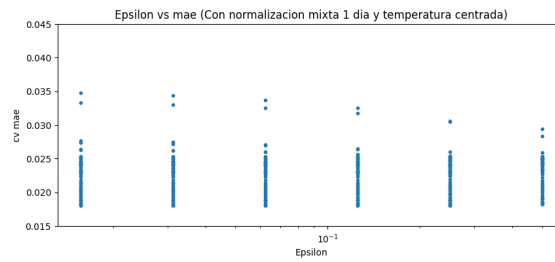
(b) Con normalización por aproximación a día laborable



(c) Con normalización por semana previa



(d) Con normalización mixta 7 días



(e) Con normalización mixta 1 día

Figura B.15: Hiperparametrización de SVR con temperatura centrada en función de ϵ

TESTS ESTADÍSTICOS

En este anexo se presentarán las distintas tablas con los resultados de realizar el *Test rank-sum de Wilcoxon* sobre los dos modelos con menor *MAE* para cada horizonte y cada modelo de *machine learning*. Dicho test estadístico compara dos series de datos para ver si hay suficiente evidencia estadística para afirmar que la distribución que genera uno de los datos es mayor que la que genera los otros. Para ello, se estudia si la serie de datos resultado de restarle la una a la otra tiene mediana 0 (hipótesis nula) o no (hipótesis alternativa).

Horizonte	Modelo con menor <i>MAE</i>	Modelo con segundo menor <i>MAE</i>	p-valor
1	Semana previa y CTC	Mixta 7 días y CTC	0,248
2	Semana previa y CTC	Mixta 7 días y CTC	0,133
3	Semana previa y CTC	Mixta 7 días y CTC	0,090
4	Mixta 7 días y CTC	Semana previa y CTC	0,712
5	Mixta 1 día y CTC	Mixta 7 días y CTC	$7,582 * 10^{-5}$
6	Mixta 1 día y CTC	Mixta 7 días y CTC	$1,109 * 10^{-5}$

Tabla C.1: p-valores del *Test rank-sum de Wilcoxon* para *Ridge Regression*

Horizonte	Modelo con menor <i>MAE</i>	Modelo con segundo menor <i>MAE</i>	p-valor
1	Semana previa y ST	Mixta 7 días y ST	0,370
2	Semana previa y CT	Mixta 7 días y CT	0,134
3	Semana previa y CT	Mixta 7 días y CT	0,046
4	Semana previa y CT	Mixta 7 días y CT	0,079
5	Mixta 1 día y CTC	Semana previa y CT	0,460
6	Mixta 1 día y CTC	Mixta 7 días y CT	0,003

Tabla C.2: p-valores del *Test rank-sum de Wilcoxon* para *Multilayer Perceptron*

Las tablas C.1, C.2 y C.3 muestran los p-valores para *Ridge Regression*, *Multilayer Perceptron* y *SVR* respectivamente. Las siglas son las explicadas en la sección 4.4.

Finalmente, también se estudió si había suficiente evidencia estadística para afirmar que los modelos de *SVR* obtienen mejores resultados que los de *Multilayer Perceptron*. Los p-valores de dichos

Horizonte	Modelo con menor <i>MAE</i>	Modelo con segundo menor <i>MAE</i>	p-valor
1	Semana previa y CT	Mixta 7 días y CT	0,100
2	Semana previa y CT	Mixta 1 día y CT	0,803
3	Semana previa y CT	Mixta 1 día y CT	0,959
4	Mixta 1 día y CT	Semana previa y CT	0,003
5	Mixta 1 día y CT	Mixta 7 días y CT	$5,926 * 10^{-5}$
6	Mixta 1 día y CT	Mixta 7 días y CT	$8,207 * 10^{-7}$

Tabla C.3: p-valores del *Test rank-sum de Wilcoxon* para SVR

Horizonte	Mejor modelo de SVR	Mejor modelo de MLP	p-valor
1	Semana previa y CT	Semana previa y ST	0,007
2	Semana previa y CT	Semana previa y CT	0,073
3	Semana previa y CT	semana previa y CT	0,135
4	Mixta 1 día y CT	Semana previa y CT	$4,270 * 10^{-4}$
5	Mixta 1 día y CT	Mixta 1 día y CTC	$5,043 * 10^{-4}$
6	Mixta 1 día y CT	Mixta 1 día y CTC	0,007

Tabla C.4: p-valores del *Test rank-sum de Wilcoxon* para SVR frente a MLP

contrastes se pueden apreciar en la tabla C.4.