UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

TRABAJO FIN DE MÁSTER

Predicción de Energía Fotovoltaica a partir de Ensembles NWP

Máster de Investigación e Innovación en Tecnologías de la Información y la Comunicación

Autor: Castillo Prieto, Edurne
Tutor: Dorronsoro Ibero, José R.

FECHA: Septiembre, 2018

Índice general

Resumen

En el mundo actual se están estudiando nuevas formas para rentabilizar el uso de las energías renovables de forma que puedan llegar a ser tan fiables como los combustibles fósiles. En el caso de España, donde una de las energías renovables más prolíferas es la energía solar debido a la favorable situación geográfica en la que se encuentra el país, las centrales solares han descubierto su necesidad de conocer con anticipación la producción que se va a obtener a la hora de gestionar los recursos de la central.

Sin embargo, hacer esas predicciones no es tarea fácil, ya que la energía solar posee una alta inestabilidad generada en su mayor parte por factores externos, como pueden ser las nubes.

Con esta motivación, este Trabajo de Fin de Máster (TFM) intenta dar una posible solución a esta problemática usando los ensembles meteorológicos del Centro Europeo para alcanzar una mejora en las predicciones. Para ello, además de los datos de modelo de ensembles se ha utilizado el algoritmo Support Vector Regression (SVR) en los modelos.

Abstract

In the current world, new ways are being studied to make the use of renewable energies profitable so that they can become as reliable as fossil fuels. In the case of Spain, where one of the most prolific renewable energies is solar energy due to the favorable geographical situation in which the country is located, solar power plants have discovered their need to know in advance the production that is going to be obtained when managing the resources of the plant.

However, making those predictions is not an easy task, since solar energy has a high instability generated mostly by external factors, such as clouds.

With this motivation, this Final Master's Project tries to give a possible solution to this problem using the meteorological ensembles of the European Center to reach an improvement in the predictions. To do this, in addition to the model data of the ensembles, the Support Vector Regression (SVR) algorithm has been used in the models.

Agradecimientos

En primer lugar quiero agradecer a mi tutor, José Dorronsoro y a la Cátedra UAM-IIC por darme la oportunidad de adentrarme en el mundo del Machine Learning. También agradecer a Alejandro Catalina el apoyo con la gestión de datos NWP. Por último, quiero agradecer a mi familia el apoyo mostrado a lo largo de todo este proceso, en especial a Jaime, por haber tenido la paciencia suficiente para leerse esta memoria.

Capítulo 1

Introducción

En un mundo donde la contaminación y la escasez de los combustibles fósiles están a la orden del día, las energías renovables están tomando poco a poco mayor relevancia. Cada vez se estudian nuevas formas de aprovechar los recursos que la naturaleza pone a nuestro alcance generando el menor número de residuos. Aunque la mayoría de las energías renovables son bien conocidas (hidráulica, biocombustible, geotérmica...) sin duda de las que más se habla, sobretodo en España, son la eólica y la solar. En este trabajo, se hablará de esta última.

En los últimos tiempos han proliferado las centrales solares; sin embargo, los costes de mantenimiento y la incertidumbre en la producción que se va a obtener dificultan su gestión y su viabilidad para conseguir su conexión a la red. La energía solar es inestable por naturaleza, de forma que hay que tener muchos factores en cuenta a la hora de intentar estimar la producción de una planta solar: desde las nubes, pasando por el viento hasta el polvo que pueda acumularse en las placas son variables determinantes en la producción de energía. Como resultado, las plantas solares necesitan a menudo ser respaldadas por los generadores auxiliares durante periodos de alta variabilidad, lo que aumenta los costes en la generación de energía eléctrica a partir de energía solar.

Una previsión anticipada y precisa de la producción de una planta solar puede facilitar su gestión y su productividad. Con el auge de los algoritmos de aprendizaje automático (Machine Learning) se han estudiado nuevas formas de abordar este problema con el objetivo de dar una predicción fiable en la que basar las gestiones de las plantas solares. Muchos expertos en el tema se han lanzado a probar toda suerte de algoritmos, entre los que destacan por su eficacia las redes neuronales y las máquinas de vectores de soporte.

En este proyecto se han marcado los objetivos de, en primer lugar, estudiar tanto lo que otros hicieron previamente en este campo como el funcionamiento de los algoritmos de aprendizaje más usados para intentar aplicar sus conclusiones a los datos de predicción proporcionados por el Centro Europeo.

De esta forma, se divide esta memoria en las siguientes partes:

- En el Capítulo 2 se explicarán los conceptos básicos de la radiación solar y la energía fotovoltaica, así como se mencionarán las técnicas más usuales de predicción y la selección de variables para los modelos.
- En el Capítulo 3 se abordarán en profundidad los algoritmos estudiados a lo largo de este trabajo, como son SVM, tanto en su versión de clasificación (SVC) como en su versión de regresión (SVR), el Descenso dual por coordenadas y Pegasos.
- En el Capítulo 4 se mostrarán los experimentos realizados y sus conclusiones.
- Finalmente, en el Capítulo 5 se hablará sobre los trabajos futuros a los que puede dar lugar este TFM y las conclusiones finales de este proyecto.

Capítulo 2

Introducción a la radiación solar y a la energía fotovoltaica

2.1. Radiación solar

La energía solar es aquella energía de origen renovable que se obtiene de aprovechar la radiación solar, es decir, el conjunto de radiaciones electromagnéticas procedentes del sol. Existen dos tipos distintos: la energía solar térmica o **termosolar** y la energía fotovoltaica [1].

La primera de ellas se basa en la concentración de la energía del sol para obtener energía térmica. Con esta energía térmica se genera vapor, que a su vez desencadenará el movimiento de una turbina que producirá electricidad. Existen dos tecnologías comerciales para obtener dicha electricidad: la tecnología de torre central (Figura 2.1.1) y la tecnología cilindroparabólica (Figura 2.1.2). En España hay una potencia instalada de aproximadamente 4GW, de los cuales 3370MW se obtienen de centrales con tecnología cilindroparabólica y 538MW en centrales con torre termosolar [1].

Por otro lado, la **tecnología fotovoltaica** (Figura 2.1.3) consiste en aprovechar

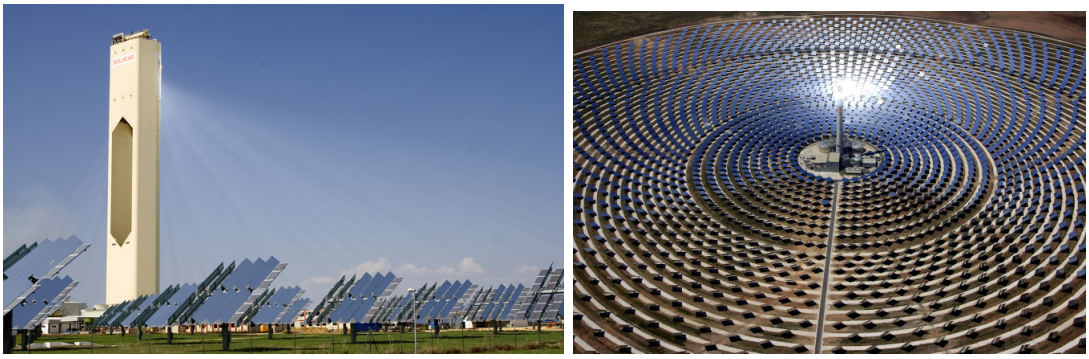


Figura 2.1.1: Central solar de torre central.
(https://commons.wikimedia.org/w/index.php?curid=2821733
y http://www.diariorenovables.com/ 2016/04/conoce-las-diferencias-entre-energia-solar-fotovoltaica-y-termosolar.html)

Figura 2.1.2: Tecnología de cilindro parabólico.

Figura 2.1.3: Planta de energía fotovoltaica.
(http: www.grupoortiz.com/negocio/concesiones/energia/proyecto-id-209/)

la radiación del Sol para generar electricidad aprovechando la reacción de algunos materiales frente a la misma. Cuando la energía solar impacta sobre ciertos materiales semiconductores produce una corriente continua. A diferencia de la termosolar, la energía fotovoltaica es capaz de producir directamente electricidad aprovechable en lugar de tener que conseguirla transformando otro tipo de energía. A finales de 2016 en España había instados 4.4GW de potencia de este tipo de energía.

2.2. Modelo Clear-Sky

Los modelos Clear-Sky estiman la radiación solar que llega a la Tierra en un día con el cielo despejado. Normalmente estiman la Irradiancia Directa Normal (DNI por sus siglas en inglés) y la Irradiancia Global Horizontal (GHI) en un cierto punto, para lo que se tienen en cuenta numerosos parámetros físicos.

La aproximación a un modelo Clear-Sky muy simplificado que se va a explicar a continuación se puede ampliar en [2]. En primer lugar se van a describir las variables. La radiación directa en un punto dado se va a denotar como $\mathcal{I}^d$, la radiación horizontal como I y el ángulo solar de indidencia o ángulo cenital, es decir, el ángulo entre los rayos solares incidentes y la vertical de un punto, como Θ. Entonces se va a tener que $I = \mathcal{I}^d \cos\Theta$, donde Θ es una función trigonométrica $\Theta = \Theta(L, D, H)$ de latitud del punto L, día D y hora H.

La constante solar, definida como la radiación en un punto dado de la atmósfera, podría ser una primera aproximación a $\mathcal{I}^d$. La radiación solar directa actual en la cima de la atmósfera varía con la distancia del Sol a la Tierra, con un máximo de $1{,}417 W/m^2$ en el perihelio (cuando el Sol está más cerca de la Tierra), un mínimo de $1{,}325 W/m^2$ en el afelio (cuando el Sol está más lejos de la Tierra) y un valor medio I_A de $1{,}370 W/m^2$. Denotando N como el número del día, la siguiente fórmula sería una buena aproximación para la radiación solar directa I_S [2]:

$$I_S = 1{,}370 \left(1 + 0{,}034 \cos\left(2\pi \frac{N-3}{365}\right)\right) \ ; \tag{2.2.1}$$

recordamos que el afelio es aproximadamente el 3 de Enero. Sin embargo, es necesario tener en cuenta la longitud del recorrido de los rayos del Sol en la atmósfera para derivar $\mathcal{I}^d$. Esta longitud relativa se describe en términos de masa de aire A. Cuando el ángulo cenital Θ es 0 y el Sol está directamente vertical, la masa de aire se toma como 1. Cuando Θ se incrementa, también lo hace la longitud del recorrido y, por tanto, la masa de aire. La fórmula empírica de Kasten-Young da una aproximación para A:

$$A = \frac{1}{\cos\Theta + 0{,}50572(96{,}07995 - \Theta)^{-1{,}6364}} \ , \tag{2.2.2}$$

con Θ dado en grados. Para derivar la radiación directa $\mathcal{I}^d$, Meinel propuso la siguiente fórmula para combinar la constante solar y la masa del aire:

$$\mathcal{I}^d = 1{,}353 \times 0{,}7^{A^{0{,}678}} \ . \tag{2.2.3}$$

Teniendo en cuenta todas las fórmulas presentadas anteriormente, se puede obtener la radiación horizontal directa Clear-Sky I que finalmente depende solo de Θ, es decir, de latitud, día y hora (aunque es necesario ajustar muchos otros parámetros porque dependen fuertemente de las condiciones locales).

Existen muchos modelos Clear-Sky distintos con niveles de complejidad dispares. El que se ha usado en el desarrollo de este proyecto ha sido el implementado en la librería *pvlib* de Python [3]. Para conseguir el dataset de estimaciones Clear-Sky para la Península con una resolución de 0.5^o en los años 2013, 2014 y 2015, se ha necesitado la información de los geopotenciales proporcionada por el modelo orográfico del ECMWF, para cada uno de los puntos que componen la rejilla definida anteriormente. En total se han calculado 522 puntos.

La función de *pvlib* usada durante este proceso proporcionaba tres medidas diferentes para cada punto, de las cuales se ha elegido la Irradiancia Global Horizontal. Dentro del desarrollo del proyecto, el conjunto de datos explicado en esta sección se ha usado para realizar una interpolación de formato trihorario a formato horario, tal y como se explicará más adelante.

2.3. Predicción de energía fotovoltaica

La energía fotovoltaica se ha vuelto un factor clave en algunos mercados eléctricos. La producción de este tipo de energía depende principalmente de la cantidad de irradiancia solar global incidente en los paneles, pero la irradiancia no es uniforme en el tiempo. El objetivo principal por el cual se requiere precisión de las predicciones de energía solar es reducir las incertidumbres relacionadas con este tipo de fuente de energía variable, para gestionar la red de forma más sencilla, eficiente y segura.

La predición de energía fotovoltaica producida se ve afectada por muchos factores. Para poner un poco en contexto se van a definir ciertos elementos a tener en cuenta a la hora de estudiar un modelo. En primer lugar, el máximo de potencia producida está definida como:

$$P_R = \eta S I[1 - 0{,}05(t_0 - 25)] \ , \tag{2.3.1}$$

donde η representa la eficiencia de conversión de la placa, S es la superficie de la placa, I es la radiación solar y t_0 es la temperatura del aire. De esta fórmula se llega a la conclusión de que las variables relevantes independientes de la central fotovoltaica van a estar relacionadas con la radiación solar y la temperatura.

Además, también es importante conocer los distintos horizontes de predicción. Cada uno indica el rango de tiempo en el que quieren predecir. Son los siguientes:

- Muy corto plazo: abarca desde 2 o 3 horas hasta 12 horas a partir del momento actual.
- Corto plazo: comprende entre las 12 y las 72 horas a partir del momento actual.
- Medio plazo: abarca desde 3 hasta 15 días a partir del momento actual.
- Largo plazo: comprende desde varios meses hasta aproximadamente un año a partir del momento actual.

En este proyecto el horizonte de predicción es a corto plazo, ya que lo que se hace es predecir todas las horas del año 2015 a partir de predicciones NWP diarias.

Respecto a la predicción de la producción de una planta solar existen dos aproximaciones principales: indirecta y directa. Las predicciones indirectas primero predicen la irradiancia solar y después obtienen la energía producida usando un modelo físico de la planta, mientras que las predicciones directas calculan la producción de energía en la planta directamente [4].

Dentro de estas dos aproximaciones hay tres técnicas principales:

- Modelos físicos.
- Modelos estadísticos.
- Modelos híbridos.

Los **modelos físicos** usan ecuaciones analíticas para modelar el sistema de energía fotovoltaica. La principal ventaja de estos modelos es que no necesitan datos históricos y, por tanto, permiten obtener la producción de energía de una planta antes de su construcción. Algunos ejemplos de modelos físicos son los siguientes [5] :

- Modelo del cielo basado en imágenes: se basa en el análisis de las estructuras de las nubes durante un periodo dado.
- Modelos basados en Predicciones Numéricas del Tiempo Atmosférico (NWP por sus siglas en inglés). Son capaces de predecir el porcentaje de irradiancia solar y de cobertura de nubes basados en modelizaciones numéricas dinámicas de la atmósfera. Este modelo se ampliará en la Sección 2.4.

De forma contraria a los modelos físicos, los **modelos estadísticos** no necesitan información del sistema para modelarlo, pero sí que necesitan datos históricos en los que basarse para poder realizar predicciones precisas. La selección de un conjunto de datos de entrenamiento se vuelve algo crucial para desarrollar un modelo preciso. Dentro de estos modelos, existen dos tipos:

- Modelos regresivos: este conjunto de técnicas estiman la relación entre una variable dependiente y algunas variables independientes, llamadas predictores.
- Técnicas de inteligencia artificial: las más usadas son las redes neuronales, pero también se han hecho estudios con KNN, SVM/SVR y Random Forest.

Algunos ejemplos concretos de modelos estadísticos son los siguientes [5]:

- Persistencia: el futuro objetivo de predicción es la media de las últimas T medidas.
- ARMA (Autoregressive Moving Average): modelo de serie temporal. Se aplica a los datos de series temporales auto correlacionadas.
- ARIMA (Autoregressive Integrated Moving Average): también es un modelo de serie temporal, pero fue desarrollado para los procesos aleatorios no estacionarios.
- ARMAX (Autoregressive–moving-average model with exogenous inputs model): es el modelo ARMA pero considerando también variables exógenas.

Por último, los **modelos híbridos** combinan los otros dos tipos de modelos anteriormente mencionados.

En este TFM se han usado técnicas de inteligencia artificial, concretamente se ha aplicado el algoritmo de Support Vector Regression (SVR), el cual se explicará en detalle en el Capítulo 3.

2.4. Predicciones Numéricas del Tiempo Atmosférico

Como se ha explicado anteriormente, en este proyecto se va a usar un modelo físico basado en predicciones numéricas del tiempo atmosférico (NWP). Los datos necesarios para realizar los experimentos han sido proporcionados por el Centro Europeo para Predicción Meteorológica a Medio Plazo (**ECMWF**). Este centro es una organización intergubernamental dedicada por una parte a la investigación y por otro lado a la oferta de Predicciones Numéricas del Tiempo atmosférico (NWP, por sus siglas en inglés) [6].

Los **NWP** son modelos computacionales que calculan numéricamente los cambios que se producen en la atmósfera, describiendo las condiciones meteorológicas actuales y cómo cambian a lo largo del tiempo mediante ecuaciones físicas. [7].

El Centro Europeo realiza sus predicciones con un modelo NWP desarrollado por ellos mismos denominado Sistema de Predicción Meteorológica Integrado (**IFS** por sus siglas en inglés) [8]. En términos generales, modelan las dinámicas de la atmósfera y los procesos físicos que ocurren en ella, como la formación de nubes, teniendo en cuenta en todo momento que la atmósfera es caótica.

Este TFM se ha desarrollado con los datos proporcionados por el modelo determinista o High Resolution (HR) y por el modelo Ensemble Prediction System. El Centro Europeo obtiene sus predicciones meteorológicas a partir del modelo IFS comentado anteriormente, en el que introducen unos valores como condiciones iniciales de la atmósfera.

En el caso del modelo determinista, las predicciones se obtienen como resultado de aplicar la mejor estimación para las ecuaciones del modelo de predicción y sus condiciones iniciales. Además, se obtiene en España con resolución 0.1º, lo que equivalea unos 9Km de separación horizontal entre puntos. En términos generales, se puede considerar como la predicción meteorológica más precisa.

Sin embargo, aunque el modelo determinista proporciona una predicción bastante exacta, las predicciones meteorológicas están sujetas a un grado considerable de incertidumbre. Por este motivo, el Centro Europeo desarrolló su **modelo de ensembles NWP**, denominado Ensemble Prediction System [9].

Este sistema genera predicciones meteorológicas a partir de la alteración de las condiciones iniciales de la atmósfera dadas en el IFS. Concretamente, el Centro Europeo proporciona 51 posibles predicciones, donde una de ellas, la conocida como ensemble de control, se obtiene a partir de las condiciones iniciales medidas y el resto, de modificar o perturbar dichas condiciones. Ambos casos se obtienen con resolución 0.2º, es decir, la distancia entre puntos en España es de aproximadamente 18Km.

Además, para generar las predicciones de ensembles se usan modelos matemáticos y parámetros diferentes. Todo esto influye en la calidad final de las predicciones, de las cuales la más ajustada a la realidad debería ser la determinista, después la de control y, por último, el resto de ensembles.

2.5. Selección de variables para los modelos

Aunque el Centro Europeo proporciona los datos necesarios para realizar los experimentos, hubo que hacer primero una selección de las variables que afectaban directamente al problema aquí expuesto.

Tras la revisión de la literatura asociada, se pudo observar que muchos autores asociaban la inestabilidad de la energía solar a las condiciones climátológicas como la irradiancia solar y la temperatura del aire [10]. Algunos además, realizaron un estudio en el que descubrieron que las variables que estaban más relacionadas con lo que querían predecir (la intensidad de la radiación solar) eran la cobertura del cielo, la humedad y la posibilidad de precipitación [11].

Teniendo esto en cuenta se analizaron las diferentes variables que proporciona el Centro Europeo. Se decidió dividir las variables en tres bloques: variables de radiación, variables de temperatura y viento y variables relacionadas con la cobertura del cielo.

2.5.1. Variables de radiación

En este bloque se estudiaron las siguientes variables:

- Clear-sky solar radiation at surface (CDIR): es la radiación solar obtenida del modelo clear-sky en la superficie de la Tierra.
- Direct solar radiation: es la incidencia en un plano perpendicular a la dirección del sol.
- Surface net solar radiation (SSR): es la radiación neta en la superficie de la Tierra.
- Surface net solar radiation clear-sky (SSRC): Radiación neta en la superficie asumiendo que no hay nubes (por el modelo clear-sky).
- Surface solar radiation downwards (SSRD): Radiación solar incidente (onda corta).
- Total sky direct solar radiation at surface (FDIR): es la radiación que cae de forma directa a la superficie.

De estas seis variables iniciales se eliminaron dos, ya que de la Direct solar radiation no había datos para los años que necesitábamos y tras hacer una comparativa de comportamiento de las variables salió a la luz que SSR y SSRD se parecían mucho, así que se escogió SSR.

Tipo	Nombre	Siglas
Radiación	Clear-sky solar radiation at surface	CDIR
Radiación	Surface net solar radiation	SSR
Radiación	Surface net solar radiation clear-sky	SSRC
Radiación	Total sky solar radiation at surface	FDIR
Nubes	Total cloud cover	TCC
Temperatra	2 metre temperature	T2M
Viento	10 metre U wind component	U10
Viento	10 metre V wind component	V10

Cuadro 2.5.1: Variables escogidas para entrenar los modelos.

2.5.2. Cobertura del cielo (nubes), temperatura y viento

En este bloque se estudian las variables que no son de radiación pero que también influyen en los modelos predictivos. En primer lugar se eligió la variable Total Cloud Cover (TCC) relacionada con la cobertura del cielo, ya que es la que proporciona información sobre la cantidad de nubes que hay en el cielo en un momento determinado.

Respecto a la temperatura, la variable elegida fue la temperatura a 2 metros (T2M) porque como las placas solares se encuentran situadas a escasos metros del suelo, la temperatura que haga a esa distancia va a influir en la producción.

Por otro lado, como variables asociadas al viento se escogieron la componente U del viento (U10) y la componente V del viento (V10), ambas a 10 metros del suelo. Finalmente, en la Tabla 2.5.1 se muestra un resumen de las variables con las que se va a trabajar a lo largo de este proyecto.

Capítulo 3

Teoría de Regresión de Vectores de Soporte

En este capítulo se hará un repaso de la teoría en la que se basan las principales técnicas que se han considerado a la hora de abordar este TFM. En primer lugar se hablará sobre los conceptos básicos de regresión, dando paso a técnicas avanzadas como Support Vector Regression (SVR) y sus implementaciones sobre núcleos lineales, Liblinear y Pegasos.

3.1. Introducción al problema de regresión

La regresión es una técnica cuyo objetivo es representar mediante alguna función el patrón que siguen los datos. Su forma de proceder es ajustar los parámetros de dicha función para que se parezca lo más posible a los datos. El modelo más básico de regresión es la **Regresión Lineal**, que tiene la siguiente forma:

$$\hat{y} = w_0 + \sum_{i=1}^{D} x_i^T w_i \ , \tag{3.1.1}$$

donde X, cuyas filas son un conjunto de patrones x, es la matriz $N \times D$ de datos y la $\hat{y}$ es la estimación para x del objetivo o *target* Y, dado por un vector de dimensión $N \times 1$ (N: número de ejemplos, D: número de dimensiones). El término w_0 representa el sesgo (*bias*) del modelo. Este modelo asume que la función de regresión es lineal y que es una buena aproximación a la distribución de los datos, lo que es cierto en muchas ocasiones [12]. El método de estimación más popular para este modelo es el de Mínimos Cuadrados (Least Squares), que compara los resultados de la función predicha con datos reales para conocer la distancia entre ellos y minimiza la siguiente función de coste:

$$\begin{aligned} RSS(w) &= \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \ ; \\ &= \sum_{i=1}^{N} (y_i - w_0 - \sum_{j=1}^{D} x_{ij} w_j)^2 \ . \end{aligned} \tag{3.1.2}$$

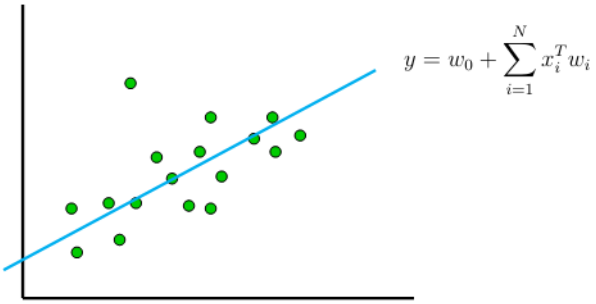


Figura 3.1.1: Representación de una regresión lineal.

Este método obtiene los coeficientes w minimizando la suma de los cuadrados de la diferencia entre el target real y el predicho.

$$\hat{w} = \arg\min \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \ . \tag{3.1.3}$$

Sin embargo, minimizar únicamente Least Squares es sensible a variaciones en los datos y además, puede dar lugar a soluciones que no sean únicas. Por este motivo, se introduce la regularización, que ayudará a solucionar estos problemas y también a controlar la complejidad del problema y evitar el *overfitting*.

La regularización en regresión lineal se introduce en el modelo llamado **Ridge Regression**, cuya función a minimizar es la siguiente:

$$y = \sum_{i=1}^{N} (y_i - w_0 - \sum_{j=1}^{D} x_{ij} w_j)^2 + \lambda \sum_{k=1}^{D} w_k^2 \ . \tag{3.1.4}$$

donde los coeficientes w se pueden calcular como

$$\hat{w}_{ridge} = \arg\min_{w} \left\{ \sum_{i=1}^{N} (y_i - w_0 - \sum_{j=1}^{D} x_{ij} w_j)^2 + \lambda \sum_{k=1}^{D} w_k^2 \right\} \ . \tag{3.1.5}$$

La solución en forma matricial sería:

$$\hat{w}_{ridge} = (X^T X + \lambda I)^{-1} X^T y \ . \tag{3.1.6}$$

Sin embargo, los datos no siempre siguen tendencias lineales y, por ello, es necesario hablar de **regresión no lineal**.

La regresión no lineal es aquella técnica de regresión cuya función a ajustar no es una recta, sino un polinomio, una exponencial o en general, modelos no lineales. La función de coste $RSS(w)$ sigue siendo la misma, solo cambia la forma de expresar $\hat{y}$.

La cuestión a resolver en este TFM es un problema de regresión con datos de alta dimensión, por lo que se han estudiado distintas técnicas de regresión, tanto

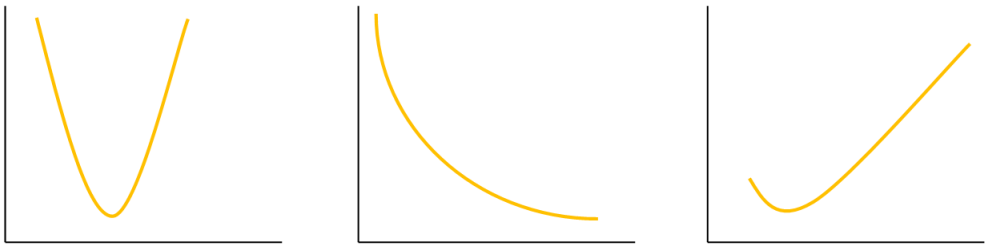


Figura 3.2.1: Ejemplos de funciones convexas.

lineales como no lineales para al final decantarse por las basadas en Support Vector Regression (SVR). Este subapartado ha dado una visión general de la regresión para poner en situación las siguientes secciones, en las que se hablará de técnicas eficaces para resolver dicho problema.

3.2. Teoría de la optimización

En esta sección se va a explicar brevemente la teoría de la optimización convexa para problemas con restricciones [13], necesaria para resolver el problema de SVR. En primer lugar se introducirá el objetivo de la optimización, las funciones convexas y, posteriormente, se entrará más en profundidad en la Teoría Lagrangiana [14].

3.2.1. Teoría de la optimización convexa

El objetivo de la teoría de la optimización es encontrar un máximo o un mínimo para una función dada, de forma que ese punto sea la mejor solución al problema que se quiere optimizar. Además, dicha función puede estar sujeta a una serie de restricciones, como el problema de optimización de SVR, del que se hablará en la sección 3.3.

Por tanto, lo que se pretende hallar es la resolución del problema en la llamada **región factible**, es decir, la región del dominio donde está definida la función y además se cumplen todas las restricciones a las que está sujeta.

El tipo de problemas más interesante para minimizar son los **problemas convexos**, que son aquellos definidos por funciones convexas. De entre todas las propiedades que poseen dichas funciones, la que más interesa en este caso es aquella que dice que toda función convexa va a tener la característica principal de poseer un mínimo, que va a ser un mínimo global. En la Figura 3.2.1 se muestran algunos ejemplos de funciones convexas.

Resolver un problema de optimización convexa implica, por tanto, que la solución a dicho problema va a ser, en general, un punto $w^* \in \mathbb{R}^n$. Además se asegura que no va a haber ningún otro punto $w \in \mathbb{R}^n$ para el que $f(w) < f(w^*)$, aunque la solución puede estar compuesta por un conjunto de puntos con valores de f idénticos. Esto quiere decir que el w mínimo no tiene que ser único y puede haber muchos puntos con el mismo valor $f(w^*)$ de la función objetivo. Entonces la solución sería un conjunto en vez de un único punto.

3.2.2. Teoría Lagrangiana

Dado el problema primal

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} f(x) \\ &t.q. \;\; g_i(x) \leq 0, \;\; i = 1, ..., m \\ &\qquad h_j(x) = 0, \;\; j = 1, ..., p \; , \end{aligned} \tag{3.2.1}$$

la Teoría Lagrangiana proporciona un problema de optimización equivalente que a menudo tiene propiedades complementarias y es más fácil de resolver. El problema de optimización anterior trata de encontrar un mínimo de la función $f(x)$ que cumpla las restricciones impuestas por $g_i(x)$ y $h_j(x)$.

Un elemento muy importante de la Teoría Lagrangiana es la función Lagrangiana o Lagrangiano. Su misión consiste en incluir en una única expresión la función a minimizar junto con sus restricciones. De esta forma, el problema se vuelve más manejable. El Lagrangiano asociado a (3.2.1) es el siguiente:

$$\mathcal{L}(x, \alpha, \beta) = f(x) + \sum_{i=1}^{m} \alpha_i g_i(x) + \sum_{i=1}^{p} \beta_i h_i(x) \; , \tag{3.2.2}$$

donde x es la variable primal y α con $\alpha_i \geq 0$ y β los denominados *multiplicadores de Lagrange*. Los multiplicadores asociados a cada una de las restricciones se pueden entender como el coste asociado a su incumplimiento.

Un punto cualquiera será factible para el problema primal siempre y cuando cumpla las restricciones del mismo. En este caso, las restricciones son $g_i(x) \leq 0, \;\; i = 1, ...m$ y $h_i(x) = 0, \;\; i = 1, ...p$. Sin embargo, se hace complicado sortear cada una de estas condiciones, en especial en problemas de mayor dimensión. Por este motivo se buscó un problema de optimización equivalente con menos restricciones cuya solución pudiera ser la misma que la del problema original. De esta forma surgió el problema dual, que es un problema equivalente al primal hasta el punto de que si se dan las condiciones idóneas pueden ser totalmente intercambiables. El problema dual es el siguiente:

$$\max_{\alpha \geq 0, \beta} \Theta(\alpha, \beta) \; , \tag{3.2.3}$$

donde

$$\Theta(\alpha, \beta) = \min_{x} \mathcal{L}(x, \alpha, \beta) \; . \tag{3.2.4}$$

Los distintos valores de α y β serán factibles en el dual siempre que cumplan la restricción asociada a dicho problema, que en este caso es $\alpha_i \geq 0,\ i = 1, ..., m$.

A continuación, se van a ver las relaciones que tienen ambos problemas (primal y dual) entre sí.

Teorema 3.2.1. *(Teorema de la Dualidad Débil) Sea x^* una solución óptima del problema primal con valor $p^* = f(x^*)$; entonces se cumple que*

$$\min_x \mathcal{L}(x, \alpha, \beta) \ \leq p^* \tag{3.2.5}$$

Esto quiere decir que el problema dual con unos α y β factibles proporciona un límite inferior para p^*.

Las diferencias que pueda haber entre las soluciones de los problemas primal y dual se conocen como *duality gap*. Lo que interesa es que este *duality gap* sea inexistente, para que ambos problemas lleguen a las mismas soluciones y, por tanto, se pueda usar uno u otro indistintamente. Esto es lo que dice la Definición de la Dualidad Fuerte enunciado a continuación:

Definición 3.2.1. *(Dualidad Fuerte) Se dice que una pareja de problemas primal-dual poseen una dualidad fuerte si los valores de las soluciones óptimas x^*, α^*, β^* de ambos problemas, definidas como $p^* = f(x^*)$ son iguales. Es decir, se cumple*

$$\min_x \mathcal{L}(x^*, \alpha^*, \beta^*) = p^* \tag{3.2.6}$$

El siguiente teorema especifica las características que deben cumplir las soluciones óptimas de un problema de optimización general cuando el gap dual es 0.

Teorema 3.2.2. *(Kuhn-Tucker) Dado un problema de optimización convexo como el representado en (3.2.1) con gap dual 0, las condiciones necesarias y suficientes para que un punto w^* sea una solución óptima son que tienen que existir un α^* y β^* tal que:*

$$\begin{aligned}
\frac{\partial \mathcal{L}(w^*, \alpha^*, \beta^*)}{\partial w} &= 0, \\
\frac{\partial \mathcal{L}(w^*, \alpha^*, \beta^*)}{\partial \beta} &= 0, \\
\alpha_i^* g_i(w^*) &= 0, \quad i = 1, ..., m, \\
g_i(w^*) &\leq 0, \quad i = 1, ..., m, \\
\alpha_i^* &\geq 0, \quad i = 1, ..., m.
\end{aligned} \tag{3.2.7}$$

Estas condiciones son conocidas como de Karush-Kuhn-Tucker (KKT).

3.3. Support Vector Regression (SVR)

Originalmente las Support Vector Machines (SVM) solo resolvían problemas de clasificación en los que se quería separar dos clases. Posteriormente se propuso una

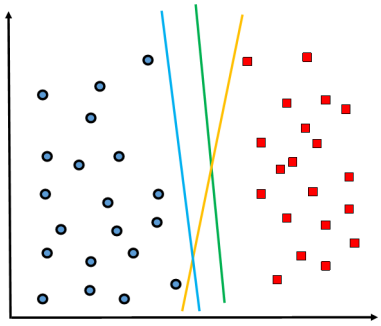


Figura 3.3.1: Posibles hiperplanos separadores para dos clases.

extensión de este método, que tan buenos resultados obtenía, para la resolución de problemas de regresión.

En este apartado se va a introducir primero Support Vector Classifier (SVC) [12] [13], para luego mostrar las diferencias introducidas en Support Vector Regression (SVR) [13].

3.3.1. SVC: Método del margen estricto

SVC se basa en el método del máximo margen. Dicho método tiene dos versiones: el método del margen estricto y el método del margen suave, de las cuales se usa la segunda de ellas. En este primer contacto con el método del margen se hablará de su versión más sencilla.

Ambas versiones coinciden en la asunción de que que existe un hiperplano que separa los ejemplos positivos y negativos dejando la mayor distancia posible entre ambas clases. En la Figura 3.3.1 se ilustran varios posibles hiperplanos separadores.

La función que define al hiperplano no es más que la ecuación del mismo. Por ese motivo, y atendiendo a la disposición de cada una de las clases denotadas en la Figura 3.3.1, se puede afirmar que si se tiene un conjunto de patrones $S : \{x_1, x_2, ..., x_n\}$ con una etiqueta $y = \{-1, +1\}$ asociada, entonces se quiere que:

$$\begin{aligned} x_i^T w + w_0 \geq +1 \;\; si \;\; y_i = +1 \\ x_i^T w + w_0 \leq -1 \;\; si \;\; y_i = -1 \; ; \end{aligned} \tag{3.3.1}$$

es decir, los puntos situados por encima del hiperplano pertenecen a la clase $y = +1$ y los que están por debajo a la clase $y = -1$. En esta primera aproximación del método del margen no se permite que ningún punto se encuentre dentro del margen ni mal clasificado, por lo que solo se podrían resolver problemas linealmente separables. Más adelante se retomará esta idea.

Si ambas ecuaciones definitorias de las clases se juntaran en una sola, quedaría de la siguiente forma:

$$y_i(x_i^T w + w_0) - 1 \geq 0 \;\; \forall i \; . \tag{3.3.2}$$

Sabiendo que el problema a resolver por SVC es la maximización del margen teniendo en cuenta la correcta clasificación de todos los puntos, la formulación sería la siguiente [14]

$$\min_{w,w_0} \frac{1}{2}||w||_2^2 \quad t.q. \quad y_i(x_i^T w + w_0) - 1 \geq 0 \ \ \forall i \ , \tag{3.3.3}$$

donde $||w||_2^2$ es el inverso de la anchura del margen que se quiere maximizar.

Tal y como se ha comentado antes, este problema no admite puntos mal clasificados y, por tanto, se restringe a problemas lineales. Sin embargo, este tipo de porblemas no son los más comunes en el mundo real. Esto quedó demostrado en el Teorema de Cover. Este teorema indica la cantidad de dicotomías separables que existen para un problema cuyos puntos se encuentren en disposición general, es decir que no haya $D+1$ patrones en un hiperplano $D-1$ dimensional.

Entonces, si los puntos se encuentran en disposición general, el número de dicotomías $L(N, D)$ linealmente separables es [15]:

$$f(x) = \left\{ \begin{array}{ll} 2^N & si \quad N \leq D+1 \\ & \\ 2\sum_{i=0}^{D} \left(\begin{array}{c} N-1 \\ i \end{array} \right) & si \quad N \geq D+1 \end{array} \right\} \ . \tag{3.3.4}$$

Como el número de dicotomías es 2^N, en la práctica si $N \gg D$ se puede demostrar que

$$\frac{2\sum_{i=0}^{D} \left(\begin{array}{c} N-1 \\ i \end{array} \right)}{2N} \to 0 \ . \tag{3.3.5}$$

Esto implica que va a ser difícil encontrar un problema linealmente separable, lo que conlleva que haya que buscar alguna forma de resolver problemas no linealmente separables. Más adelante se hablará sobre el método del núcleo (kernel).

3.3.2. SVC: Método del margen suave

Volviendo al método del margen que se había explicado para SVC, ha quedado demostrado que existen muy pocos problemas linealmente separables y que este método no va a tener gran utilidad en problemas reales. La primera solución que se propuso fue la introducción de unas variables cuyo objetivo es relajar la restricción de (3.3.3) para que se permita tener algunos ejemplos mal clasificados. Estas variables se llaman variables "slack", y controlan de alguna manera la cantidad de ejemplos mal clasificados que se van a permitir. La restricción del problema SVC modificada sería:

$$y_i(x_i^T w + w_0) \geq 1 - \xi_i \ \ \forall i \ , \tag{3.3.6}$$

donde $\xi_i \geq 0$ son las variables slack que van a hacer posible que un punto se encuentre dentro del margen ($0 \leq \xi_i \leq 1$) o que se clasifique mal ($\xi_i > 1$).

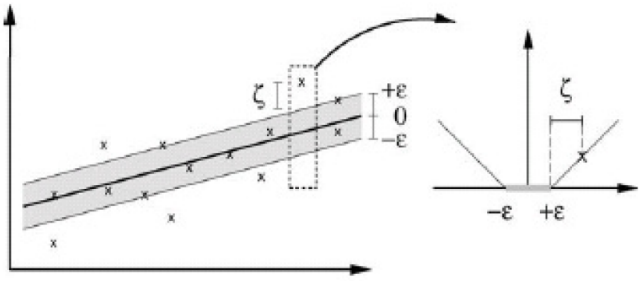


Figura 3.3.2: Representación del "tubo" formado por el error "ϵ-insensitivo". (Imagen obtenida de [14]).

Teniendo esto en cuenta, la expresión del problema SVC se modifica de la siguiente forma:

$$\min_{w,w_0} \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{N} \xi_i \ , \tag{3.3.7}$$

$$t.q. \quad y_i(x_i^T w + w_0) \geq 1 - \xi_i \ \ \forall i \ ,$$
$$\xi_i \geq 0 \ .$$

Como se puede ver, se ha añadido el término de penalización $C\sum_i \xi_i$, formado por $\sum_i \xi_i$, que actúa como cota superior para el número de errores de clasificación y por $C > 0$, que controla la compensación entre la maximización del margen y la minimización del error.

Partiendo de la expresión (3.3.7), se va a proceder a explicar las diferencias de este problema con SVR.

3.3.3. SVR

Como se ha comentado anteriormente, SVC se apoya en el método del margen. Sin embargo, para poder enfrentarse a problemas de regresión se definió el problema SVR como una ampliación de SVC en la que se añadía una nueva función de error, el llamado error "ϵ-insensitivo", cuyo propósito es ignorar los errores cometidos menores que un umbral ϵ.

El problema de SVR se define como:

$$\min_{w,w_0} \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{N} V_\epsilon(y_i - x^T w - w_0) \ , \tag{3.3.8}$$

En la fórmula anterior, la nueva función de error está denotada como V_ϵ, y se define de la siguiente manera:

$$V_\epsilon(y_i - f(x_i)) = \begin{cases} 0 & si \quad |y_i - f(x_i)| < \epsilon \ ; \\ |y_i - f(x_i) - \epsilon| & si \quad |y_i - f(x_i)| \geq \epsilon \ . \end{cases} \tag{3.3.9}$$

Esta función de error genera un "tubo"de radio ϵ alrededor de la función, de tal forma que únicamente los puntos que se sitúen fuera del tubo serán penalizados en el ajuste de la función.

Este tubo es semejante al margen de SVC. La diferencia es que ahora, en vez de separar los puntos en dos clases, se intenta agruparlos dentro del tubo. Además, el error medido desde el tubo al punto considerado erróneo se denota como ξ, exactamente igual que ocurría en SVC con las variables slack.

Por tanto, incluyendo la definición del error "ϵ-insensitive" en la formulación inicial de SVR se obtiene un problema muy parecido al de SVC:

$$\min_{w,w_0,\xi,\xi'} \frac{1}{2}||w||^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i') \ ; \tag{3.3.10}$$

$$\begin{aligned} t.q. \quad & x_p^T w + w_0 - y_i \leq \epsilon + \xi_i' \ , \\ & y_i - x_i^T w - w_0 \leq -\epsilon - \xi_i \ , \\ & \xi_i, \xi_i' \geq 0 \ . \end{aligned}$$

ξ y ξ' representan los errores cometidos por encima o por debajo del tubo. Presentando el problema de esta manera se puede fácilmente observar que se trata de un problema de optimización convexo con restricciones y, por tanto, se puede resolver usando la Teoría Lagrangiana vista en la sección 3.2.2.

Entonces, (3.3.10) sería el problema primal. Para llegar hasta la formulación dual con menos restricciones hay que definir primero el Lagrangiano del problema primal:

$$\begin{aligned} \mathcal{L}(w, w_0, \xi, \xi', \alpha, \beta, \gamma, \delta) =& \frac{1}{2}||w||^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i') - \sum_{i=1}^{N}\alpha_i(x_i^T w + w_0 - y_i + \xi_i + \epsilon) \\ & + \sum_{j=1}^{N}\beta_j(x_j^T w + w_0 - y_j - \xi_j' - \epsilon) - \sum_{i=1}^{N}\gamma_i\xi_i - \sum_{j=1}^{N}\delta_j\xi_j' \ . \end{aligned} \tag{3.3.11}$$

Partiendo de esta expresión, la función dual se obtiene sustituyendo en el Lagrangiano primal el resultado de derivar parcialmente $\mathcal{L}$ con respecto a w, w_o, ξ y ξ' e igualarlos a cero. El resultado de esta operación es:

$$\Theta(\alpha, \beta) = \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \beta_i)(\alpha_j - \beta_j)x_i^T x_j + \epsilon\sum_{i=1}^{N}(\alpha_i + \beta_i) - \sum_{i=1}^{N} y_i(\alpha_i - \beta_i) \ . \tag{3.3.12}$$

A partir de esta función ya se tiene el problema dual a resolver. Como se busca un máximo, se va a hacer que el problema se siga considerando estrictamente convexo al sustituirlo por un mínimo cambiado de signo.

$$\max_{\alpha,\beta} \Theta(\alpha, \beta) = -\min_{\alpha,\beta} \Theta(\alpha, \beta) \ ; \tag{3.3.13}$$

$$\begin{aligned} & t.q.\ 0 \leq \alpha_i\ ,\ \beta_j \leq C\ , \\ & \sum_{i=1}^{N} \alpha_i = \sum_{j=1}^{N} \beta_j\ . \end{aligned} \tag{3.3.14}$$

Las condiciones KKT para este problema son, para $1 \leq i, j \leq N$:

$$\begin{aligned} w^* &= \sum_i (\alpha_i^* - \beta_i^*) x_i \ \text{ con } \ \alpha_i^* \beta_i^* = 0\ ; \\ 0 &= \alpha_i^* (x_i^T w^* + w_0^* - y_i^* + \xi_i^* + \epsilon)\ , \\ 0 &= \beta_j^* (x_j^T w^* + w_0^* - y_j^* - \xi_j^{'*} - \epsilon)\ , \\ 0 &= (C - \alpha_i^*)\xi_i^*\ , \\ 0 &= (C - \beta_j^*)\xi_j^{'*}\ . \end{aligned} \tag{3.3.15}$$

Si se toma como referencia la Figura 3.3.2, se puede ver que va a haber dos tipos distinguibles de puntos:

- Los puntos que cumplan $0 < \alpha_i^* < C$ y , por tanto, $\xi_i^* = 0$ y $x_i^T w^* + w_0^* - y_i = -\epsilon$, que son los que se encontrarán sobre la línea del margen superior.

- Los puntos que se encuentren sobre la línea del margen inferior, que cumplirán $0 < \beta_j^* < C$ y ,entonces, $\xi_j^{'*} = 0$ y $x_j^T w^* + w_0^* - y_j = \epsilon$.

Los puntos definidos anteriormente son aquellos que se encuentran directamente sobre una de las dos fronteras que definen el tubo de ancho ϵ. Sin embargo, los puntos que van a influir en la función son todos aquellos que se encuentren en los hiperplanos o fuera del tubo . Dichos puntos son los vectores de soporte mencionados con anterioridad en SVC y los del hiperplano.

El parámetro w_0^* se puede despejar de dos formas distintas, una para los parámetros α y otra para los parámetros β. A continuación, se va a mostrar el desarrollo para α_i. En primer lugar, se escoge un α_i^* que cumpla la restricción de caja $0 < \alpha_i^* < C$. Conociendo este dato, la condición KKT

$$0 = (C - \alpha_i^*)\xi_i^*\ , \tag{3.3.16}$$

implicaría que $\xi_i^* = 0$, ya que $(C - \alpha_i^*)$ es distinto de cero. Con esta nueva información, se usa otra de las condiciones KKT, de la cual se puede despejar w_0^*.

$$\begin{aligned} 0 &= \alpha_i (x_i^T w^* + w_0^* - y_i^* + \xi_i^* + \epsilon)\ , \\ &\Rightarrow 0 = x_i^T w^* + w_0^* - y_i^* + \epsilon\ ; \\ &\Rightarrow w_0^* = y_i^* - x_i^T w^* - \epsilon\ . \end{aligned} \tag{3.3.17}$$

De la misma forma se obtendría la ecuación de w_0^* para los parámetros β, de manera que

$$w_0^* = \begin{cases} y_i^* - x_i^T w^* - \epsilon & si \quad 0 < \alpha_i^* < C\ ; \\ y_j^* - x_j^T w^* + \epsilon & si \quad 0 < \beta_i^* < C\ . \end{cases} \tag{3.3.18}$$

Finalmente, el modelo que se obtiene es el siguiente:

$$f(x) = w_0^* + \sum_{i=1}^{N}(\alpha_i^* - \beta_j^*)x_i^T x \ . \tag{3.3.19}$$

Se observa que solo se necesitan calcular productos escalares para construir un modelo SVR y aplicarlo.

3.3.4. Kernel SVR

Tal y como se comentó con anterioridad en el Teorema de Cover, lo normal es que los problemas no sean linealmente separables y que $N \gg D$. Sin embargo, si se pudiera aumentar la dimensión de forma que $D \gg N$, el problema podría tratarse como linealmente separable. Esto quiere decir que si fuera posible encontrar una función Φ capaz de transformar un espacio de características en otro con dimensión mayor, se podría resolver cualquier problema de forma lineal en ese nuevo espacio.

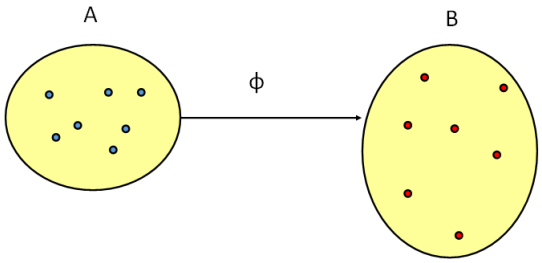


Figura 3.3.3: Representación del cambio del espacio A al espacio B mediante la función ϕ.

Sin embargo, esta función no se puede calcular fácilmente, pero si se encuentra una función $K(x, x')$ que calcule el producto escalar $\phi(x) \cdot \phi(x')$, no será necesario calcularla. Dicha función se conoce con el nombre de núcleo (kernel), y ha de cumplir:

$$K(x, x') = \langle \phi(x), \phi(x') \rangle \ . \tag{3.3.20}$$

Por tanto, para que el problema dual pueda resolver casos no linealmente separables mediante el truco del kernel, su expresión se modifica de la siguiente manera:

$$f(x) = \beta_0 + \sum_{p}(\alpha_p^* - \beta_q^*)k(x_p, x) \ . \tag{3.3.21}$$

Este problema se resolvería de la misma forma que el lineal visto con anterioridad cambiando $x_i^T x_j$ por $K(x_i, x_j)$.

A continuación se muestran los principales tipos de kernels.

Kernel Lineal

$$K(x_i, x_j) = x_i^T x_j \ . \tag{3.3.22}$$

Kernel Polinómico

$$K(x_i, x_j) = (x_i^T x_j + 1)^n \ . \tag{3.3.23}$$

Kernel Gaussiano

$$K(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) \ . \tag{3.3.24}$$

3.4. Algoritmos para SVR

Cuando se habló anteriormente del Teorema de Cover en la Fórmula 3.3.4, quedó demostrado que si en el conjunto de datos que se fuera a usar $N \gg D$ era difícil que el problema se pudiera resolver de forma lineal y por eso se introdujo el kernel trick. Sin embargo, si se tiene un conjunto de datos en el cual $D \gg N$, se pueden usar métodos lineales explícitos para intentar aproximar una solución al problema. Para simplificar la exposición, en lo que sigue nos limitaremos al caso lineal.

Por ese motivo, una vez visto cómo funciona SVR, se va a hablar de varias formas de implementarlo. Y no se puede hablar de algoritmos de SVM sin hacer mención especial a Sequential Minimal Optimization (SMO), la implementación más conocida de este método y que también es extensible a SVR (Véase [16] [17] [18]).

En general, dentro de los algoritmos de SVR se identifican dos tendencias claramente diferenciadas: la que se centra en resolver el problema dual y la que se centra en resolver el problema primal. En esta sección se van a explicar tres algoritmos distintos, de la primera vertiente SMO y Descenso Dual por Coordenadas [19][20][21] y de la segunda Pegasos [22]. Los dos primeros se van a explicar en cierta profundidad, mientras que Pegasos se repasará muy brevemente.

3.4.1. SMO

SMO es una implementación para resolver numéricamente el problema convexo presentado tanto por SVC como por SVR. Se centra en elegir una buena dirección de descenso para asegurar la factibilidad de las actualizaciones de los parámetros.

La siguiente formulación engloba a ambos problemas.

$$\min_{\beta,\beta_0,\xi} P(\beta, \beta_0, \xi) = \frac{1}{2}||\beta||^2 + C\sum_i \xi_i \ ; \tag{3.4.1}$$

$$t.q.\ y_i(x_i^T\beta + \beta_0) \geq p_i - \xi_i, \ \forall i \ ;$$
$$\xi_i \geq 0, \ \forall i \ ,$$

Para que esta fórmula sea la de SVC solo hace falta que $p_i = 1$ y que y_i represente a las etiquetas de clase, es decir:

$$\min_{\beta,\beta_0,\xi} P(\beta,\beta_0,\xi) = \frac{1}{2}||\beta||^2 + C\sum_i \xi_i \ ; \tag{3.4.2}$$

$$\begin{aligned} t.q.\ y_i(x_i^T\beta + \beta_0) &\geq 1 - \xi_i,\ \forall i \ , \\ \xi_i &\geq 0,\ \forall i \ . \end{aligned}$$

Por otro lado, para obtener la formulación de SVR hay que tener dos conjuntos, uno es el conjunto original de N patrones y el otro es el conjunto original concatenado consigo mismo con un total de $2N$ patrones. El objetivo es llegar a la restricción del problema compacto a partir de las restricciones de SVR, es decir:

$$\begin{aligned} x^T\beta + \beta_0 - y &\leq \epsilon + \xi \ , \\ x^T\beta + \beta_0 - y &\geq -\epsilon - \xi \ . \end{aligned} \tag{3.4.3}$$

La primera ecuación hace referencia a los puntos que caen fuera del tubo y por encima, y la segunda los que caen fuera del tubo y por abajo. A continuación se muestran las operaciones realizadas hasta llegar a la restricción del problema compacto:

$$\begin{aligned} & x^T\beta + \beta_0 - t \geq -\epsilon - \xi \ , \\ \text{o equivalentemente: } & x^T\beta + \beta_0 \geq -\epsilon - \xi + t \ . \\ \text{Por tanto, si } & p = -\epsilon + t \ , \\ \text{entonces } & y(x^T\beta + \beta_0) \geq -\epsilon + p \ . \end{aligned} \tag{3.4.4}$$

Se ha multiplicado a ambos lados por y, siendo $y = 1$. Con la otra condición se hace lo mismo, teniendo en cuenta que ahora $y = -1$:

$$\begin{aligned} & x^T\beta + \beta_0 - t \leq -\epsilon - \xi \ , \\ \text{o equivalentemente: } & x^T\beta + \beta_0 \leq -\epsilon - \xi + t \ . \\ \text{Por tanto, si } & p = -\epsilon + t \ , \\ \text{entonces } & y(x^T\beta + \beta_0) \geq \epsilon - p \ . \end{aligned} \tag{3.4.5}$$

Finalmente, resolver (3.4.1) es lo mismo que resolver su problema dual, debido a la teoría de Lagrange para la resolución de problemas convexos. El dual es ahora:

$$\min_{\alpha} f(\alpha) = \frac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j X_i X_j - \sum_i \alpha_i p_i = \frac{1}{2}\alpha Q\alpha - \alpha p \tag{3.4.6}$$

$$s.t.\ 0 \leq \alpha_i \leq C,\ \ 1 \leq i \leq N,\ \ \sum_i \alpha_i y_i = 0 \ ,$$

siendo $Q_{ij} = y_i y_j X_i \cdot X_j$.

Funcionamiento de SMO dual de SVM

Para resolver el problema, hay que tener en cuenta en todo momento dos restricciones:

- Restricción de caja: $0 \leq \alpha_i \leq C$
- Restricción de igualdad: $\sum_i \alpha_i y_i = 0$

La primera de ellas es fácil de cumplir. Simplemente hay que acotar los resultados obtenidos. Por ejemplo: Si $C = 1$, y los resultados obtenidos son (0.1, 1.2, -0.3, 0.5), aplicando la restricción de caja se llega a (0.1, 1, 0, 0.5)

Sin embargo, la restricción de igualdad no es tan fácil de sortear. Si se supone que no hay término independiente β_0 la restricción no existiría, ya que se obtiene de la derivada parcial de este parámetro. Esto es una "trampa"que puede funcionar solo en algunos problemas, así que hay que encontrar alguna forma de que se cumpla la restricción.

El problema se tiene que resolver numéricamente, es decir, iterando. Si se quiere usar el descenso por gradiente global, tanto el tiempo como el coste computacional asociados van a ser considerables. Este es el motivo por el que se plantea SMO. Es un algoritmo similar al descenso por gradiente que actualiza dos parámetros simultáneamente, en lugar de todos, como hacía el descenso por gradiente. Con este cambio, la complejidad del algoritmo se ha visto reducida sustancialmente. Para mejorar también el tiempo que tarda en converger, SMO elige una dirección de descenso más sencilla que alcanza antes la solución óptima.

Los parámetros son actualizados por parejas en lugar de individualmente para poder cumplir la restricción de igualdad. Si se tienen dos α, se pueden ajustar para que se compensen entre sí y que la restricción se cumpla. De esta forma se ha solucionado el problema y tanto la restricción de caja como la de igualdad se cumplen. A continuación, se dan algunos detalles sobre SMO.

Dirección de descenso factible

Teniendo definida la iteración de SMO como:

$$\alpha^{k+1} = \alpha^k + \rho_k d^k \ , \tag{3.4.7}$$

hay que comprobar que la dirección de descenso elegida (d) sea factible y, efectivamente, de descenso.

Comprobar que es factible es lo mismo que comprobar que cumple la restricción de igualdad:

$$\sum \alpha \cdot y = 0 = y \cdot \alpha^{k+1} \ . \tag{3.4.8}$$

Si se aplica la segunda forma de escribir la restricción de igualdad a la iteración de SMO, quedaría:

$$y \cdot \alpha^{k+1} = y \cdot (\alpha^k + \rho_k d^k) = y \cdot \alpha^k + \rho d^k \cdot y \ . \tag{3.4.9}$$

Sabemos que $y \cdot \alpha^k = 0$ y, por tanto, esta parte de la ecuación no va a suponer un problema para cumplir la restricción. Por otra parte, ρ va a ser siempre un número distinto de cero, así que falta por comprobar que $d^k \cdot y$ sea cero. Esto se consigue si la dirección de descenso queda definida como:

$$d^k = y^{L_k} e_{L_k} - y^{U_k} e_{U_k} \ , \tag{3.4.10}$$

donde e_L y e_U son vectores con una única posición distinta de cero, lo que implica que solo se van a tener dos coordenadas no nulas.

Si se usa esta definición para extender el término que se quería comprobar que fuera cero se tiene que:

$$d^k \cdot y = y \cdot (y^{L_k} e_{L_k} - y^{U_k} e_{U_k}) = y^{L_k} e_{L_k} \cdot y - y^{U_k} e_{U_k} \cdot y = y^{L_k} y^{L_k} - y^{U_k} y^{U_k} = 0 \ . \tag{3.4.11}$$

De esta forma queda demostrada la factibilidad de la dirección de descenso.

Por otro lado, se sabe que la dirección de descenso óptima es el menos gradiente. Por tanto, se quiere que $d \cdot \nabla f < 0$, es decir, que sea lo más negativo posible. Como d depende de la elección de L y U:

$$d \cdot \nabla f = y^{L_k} e_{L_k} \cdot \nabla f - y^{U_k} e_{U_k} \cdot \nabla f = y^{L_k} (\nabla f)_{L_k} - y^{U_k} (\nabla f)_{U_k} \ . \tag{3.4.12}$$

Teniendo esto en cuenta, L^* y U^* se definirán como:

$$L^*, U^* = \arg\min d \cdot \nabla f = \arg\min_{L,U} \left\{ y^L (\nabla f)_L - y^U (\nabla f)_U \right\} \ . \tag{3.4.13}$$

Para conseguir minimizar la ecuación anterior, es necesario que $y^L \nabla f_L$ sea lo más pequeño posible y que $y^U \nabla f_U$ sea lo más grande posible. En resumen, se quiere que:

$$L^* = \arg\min_{p \in \mathcal{I}_\mathcal{L}} \{ y^p \nabla f(\alpha)_p \} \ , \tag{3.4.14}$$

$$U^* = \arg\max_{p \in \mathcal{I}_\mathcal{U}} \{ y^q \nabla f(\alpha)_q \} \ , \tag{3.4.15}$$

siendo $\mathcal{I}_L$ y $\mathcal{I}_U$ los conjuntos de índices permisibles para poder respetar la restricción de caja:

$$\mathcal{I}_\mathcal{L} = \mathcal{I}_\mathcal{L}(\alpha) = \{ i : (y_i = +1 \wedge \alpha_i < C) \vee (y_i = -1 \wedge \alpha_i > 0) \} \ , \tag{3.4.16}$$

$$\mathcal{I}_\mathcal{U} = \mathcal{I}_\mathcal{U}(\alpha) = \{ i : (y_i = +1 \wedge \alpha_i > 0) \vee (y_i = -1 \wedge \alpha_i < C) \} \ . \tag{3.4.17}$$

Elección de ρ

Una vez elegidos L y U solo falta elegir la tasa de descenso, ρ. Para ello partimos de la formulación dual de SMO y aplicamos las fórmulas de la iteración:

$$\begin{aligned} f(\alpha_k) &= \frac{1}{2}\alpha_k \cdot Q\alpha_k - \alpha_k \cdot p \ ; \\ f(\alpha_{k+1}) &= f(\alpha_k + \rho d_k) = \frac{1}{2}\alpha_k \cdot Q\alpha_k - \alpha_k \cdot p \\ &= \frac{1}{2}((\alpha_k + \rho d_k) \cdot Q(\alpha_k + \rho d_k)) - p \cdot (\alpha_k + \rho d_k) \\ &= \frac{1}{2}\alpha_k \cdot Q\alpha_k + \frac{1}{2}\rho^2 d_k^T \cdot Qd_k + \rho d_k \cdot Q\alpha_k - p \cdot \alpha_k - pd_k \cdot \rho \\ &= \phi(\rho) \ . \end{aligned} \tag{3.4.18}$$

Como se puede ver, esta función depende solo de ρ, ya que el resto de parámetros son conocidos. Además, como se quiere que $\phi(\rho)$ sea lo más pequeño posible, hay que resolver el problema de minimización igualando la derivada de $\phi(\rho)$ a cero, y despejando:

$$\begin{aligned} \phi'(\rho) &= \rho\alpha_k \cdot Qd_k + d_k \cdot Q\alpha_k - d_k \cdot p \ ; \\ o\ equivalentemente: \quad \rho\alpha_k^T Qd_k &= -d_k Q\alpha_k + d_k p \ , \\ \implies \rho^* &= -\frac{d_k(Qd_k - p)}{d_k^T Qd_k} \\ &= -\frac{d_k \cdot \nabla f(\alpha_k)}{\alpha_k^T Qd_k} \\ &= \frac{(y_U(\nabla f)_U) - (y_L(\nabla f)_L)}{||X_L - X_U||^2} \ . \end{aligned} \tag{3.4.19}$$

Este ρ^* se conoce como el paso sin restricciones óptimo.

Ganancia

La ganancia en SMO está definida como:

$$f(\alpha_{k+1}) - f(\alpha_k) = \frac{(\triangle\ (\alpha_k))^2}{2||X_L - X_U||^2} \ , \tag{3.4.20}$$

con $\triangle\ (\alpha_k))^2 = (-d\nabla f(\alpha))^2$, asegurando que la dirección de descenso sea sencilla a la vez que proporciona el máximo descenso posible.

Anteriormente se han elegido L y U para que el numerador de ρ fuera lo más grande posible; sin embargo, no se puede decir lo mismo del denominador. Por este motivo, el método WSS2 elige L de la forma anteriormente presentada y U de la siguiente:

$$U^* = \arg\max_U \frac{((y_U(\nabla f)_U) - (y_L(\nabla f)_L))^2}{||X_L - X_U||^2} \ , \tag{3.4.21}$$

donde $U \in \mathcal{I}_U$. Esto se denomina SMO de segundo orden (WSS2).

Actualizaciones finales

Finalmente, las actualizaciones de α sin restricciones son:

$$\begin{aligned} \alpha_{L^t}^{t+1} &= \alpha_{L^t}^t + y_{L^t}\rho_t^* \ , \\ \alpha_{U^t}^{t+1} &= \alpha_{U^t}^t - y_{U^t}\rho_t^* \ . \end{aligned} \tag{3.4.22}$$

Para verificar la restricción de caja es necesario hacer un clip sobre ρ^*, tal y como se muestra a continuación.

$$\rho_t = \begin{cases} \text{mín}\{\rho_t^*, \alpha_{L^t}^t\} & si \quad y_{L^t} = -1 \ , \\ \text{mín}\{\rho_t^*, C - \alpha_{L^t}^t\} & si \quad y_{L^t} = +1 \ , \\ \text{mín}\{\rho_t^*, C - \alpha_{U^t}^t\} & si \quad y_{U^t} = -1 \ , \\ \text{mín}\{\rho_t^*, \alpha_{U^t}^t\} & si \quad y_{U^t} = +1 \ . \end{cases} \tag{3.4.23}$$

Para finalizar, observamos que los desarrollos anteriores se pueden también aplicar sobre núcleos no lineales.

3.4.2. Descenso Dual por Coordenadas

En esta sección se va a ver un método de descenso dual por coordenadas, en primer lugar el propuesto para SVC en [20] y finalmente la extensión del mismo que hicieron Ho y Lin en [19] para SVR.

La primera versión propone un método para la resolución del problema dual de SVC. En [20] explican el método con dos funciones de pérdida distintas, pero aquí se va a hablar solamente de una de ellas, *L1-SVM*, ya que coincide con el caso de un núcleo lineal en la discusión anterior.

El problema primal de SVM con la función de pérdida L1-SVM es el siguiente:

$$\min_w \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{N}(\text{máx}(0, 1 - y_i w^T x_i)) \ , \tag{3.4.24}$$

donde $\text{máx}(0, 1 - y_i w^T x_i)$ es la forma de la función de pérdida L1-SVM.

Como se puede observar, el término independiente w_0 que aparecía en SVC y que derivaba en la restricción de igualdad, no se está teniendo en cuenta en la formulación y en lo sucesivo, no se va a trabajar con él. Por ese motivo la mencionada restricción tampoco va a aparecer en el problema dual, tal y como se puede ver a continuación:

$$\min_\alpha \frac{1}{2}\alpha^T Q\alpha - e^T\alpha \ , \quad s.t. \ 0 \leq \alpha_i \leq C, \ \forall i \ . \tag{3.4.25}$$

En la fórmula anterior, $Q_{ij} = y_i y_j x_i^t x_j$ es la matriz de núcleos. La resolución del problema anterior consiste en conseguir los parámetros α óptimos mediante actualizaciones iterativas de los mismos.

El algoritmo que se propone consiste en dos bucles anidados. El bucle externo representa la actualización del vector de parámetros α en su totalidad ($\alpha^k \to \alpha^{k+1}$)

y el bucle interno la actualización individual de cada uno de los parámetros ($\alpha_i^{k,i} \rightarrow \alpha_i^{k,i+1}$).

Para actualizar los parámetros (bucle interno) se resuelve el siguiente subproblema de una variable por cada una de los parámetros individualmente:

$$\min_d f(\alpha^{k,i}+de_i) = \frac{1}{2}Q_{ii}d^2+\bigtriangledown_i f(\alpha^{k,i})d+constante\ ,\quad t.q.\ 0 \leq \alpha_i^k+d \leq C\ , \tag{3.4.26}$$

donde $\bigtriangledown_i f$ es la componente iésima del gradiente $\bigtriangledown f$ y e_i es un vector de ceros que contiene un 1 en la posición i.

La implementación para el problema dual que se propone, consiste en usar un orden aleatorio para los sub-problemas de cada iteración. Con esto se consigue un entrenamiento muy rápido. La función (3.4.26) va a tener un valor óptimo para $d = 0$ si y solo si el gradiente proyectado es cero.

$$\bigtriangledown_i^P f(\alpha^{k,i}) = 0\ . \tag{3.4.27}$$

El gradiente proyectado se define como:

$$\bigtriangledown_i^P f(\alpha) = \begin{cases} \bigtriangledown_i f(\alpha) & si \quad 0 < \alpha_i < C\ ; \\ \min(0, \bigtriangledown_i f(\alpha)) & si \quad \alpha_i = 0\ ; \\ \max(0, \bigtriangledown_i f(\alpha)) & si \quad \alpha_i = C\ . \end{cases} \tag{3.4.28}$$

El gradiente proyectado es el encargado de que la restricción de caja, por la cual $0 < \alpha < C$, se cumpla. Por este motivo si se da $0 < \alpha < C$ no hay que actualizar α porque ya se encuentra en un valor óptimo. En ese caso el valor obtenido tras resolver el problema sería la constante de (3.4.26).

Por el contrario, si no se da la mencionada condición de optimalidad se procede a actualizar ese α concreto de la siguiente forma:

$$\alpha_i^{k,i+1} = \min\left(\max\left(\alpha_i^{k,i} - \frac{\bigtriangledown_i f(\alpha^{k,i})}{Q_{ii}}, 0\right), C\right)\ , \tag{3.4.29}$$

donde $Q_{ii} = x_i^T x_i$ y $\bigtriangledown_i f(\alpha) = (Q\alpha)_i - 1 = \sum_{j=1}^{N} Q_{ij}\alpha_j - 1 = \sum_{j=1}^{N} x_i^T x_j \alpha_j - 1$. Lo que muestra esta fórmula es que se quiere tener la máxima d (desplazamiento entre un α y su actualización) sin saltarse la restricción de caja $0 < \alpha_i < C$.

Como se está resolviendo una formulación lineal, la representación explícita de los pesos primales está dada por:

$$w = \sum_{j=1}^{N} y_j \alpha_j x_j\ . \tag{3.4.30}$$

Algoritmo 1: Descenso Dual por Coordenadas para SVC

Input: $\alpha = 0 \in \mathbb{R}^d$ y C
while α *no es óptimo* **do**
  Escoger un índice i de $\{1, ..., n\}$
  $g = y_i w^T x_i - 1$
  $$p = \begin{cases} g & si \quad 0 < \alpha_i < C \ ; \\ \text{mín}(g, 0) & si \quad \alpha_i = 0 \ ; \\ \text{máx}(g, 0) & si \quad \alpha_i = C \ . \end{cases}$$
  if $|p| \neq 0$ **then**
    $\alpha_i' \leftarrow \alpha_i$
    $\alpha_i \leftarrow \text{mín}(\text{máx}(\alpha_i - g/Q_{ii}, 0), C)$
    $w \leftarrow w + (\alpha_i - \alpha_i')y_i x_i$
  end
end

Entonces se puede calcular el gradiente como:

$$\bigtriangledown_i f(\alpha) = y_i w^T x_i - 1 \ . \tag{3.4.31}$$

Finalmente, el pseudocódigo que se ha ido explicando paso a paso se puede ver en el Algoritmo 1. Este algoritmo está implementado en el paquete software LibLinear [21].

Anteriormente se ha hablado del método de Descenso Dual por Coordenadas para SVC y a partir de ahora se va a comentar cómo hicieron los autores para poder usarlo con SRV [19]. Recordando el problema primal de SVC definido en la fórmula (3.4.24), se puede observar que para definir el problema primal de SVR se ha usado la misma notación y se ha introducido en el término de regularización el error "ϵ-insensitivo":

$$\min_w \frac{1}{2}||w||_2^2 + C \sum_{i=1}^{l} \text{máx}(|w^T x_i - y_i| - \epsilon, 0) \ . \tag{3.4.32}$$

Es importante tener en cuenta que mientras en SVC se tenían N variables duales porque el problema solo tenía una restricción para cada x, SVR va a tener $2N$ debido a que tiene dos. Para las primeras N variables de SVR se usará un conjunto de coeficientes α^+ y para el resto de variables los coeficientes α^-. Como se puede observar, hay un cambio de notación respecto a la Sección 3.3.3. Ahora los coeficientes duales pasan de denotarse $\{\alpha, \beta\}$ a $\{\alpha^+, \alpha^-\}$. Teniendo esto en cuenta, el dual se define de la siguiente forma:

$$\min_{\alpha^+, \alpha^-} \frac{1}{2}(\alpha^+ - \alpha^-)^T Q(\alpha^+ - \alpha^-) - \sum_{i=1}^{N} y_i(\alpha_i^+ - \alpha_i^-) + \epsilon \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-) \ ; \tag{3.4.33}$$

$$s.t. \ 0 \leq \alpha_i^+, \alpha_i^- \leq C, \ \forall i = 1, ..., N \ .$$

Este problema se puede escribir de forma más compacta si se agrupan los coeficientes en un vector:

$$\alpha = \begin{bmatrix} \alpha^+ \\ \alpha^- \end{bmatrix} \tag{3.4.34}$$

Por tanto, el problema dual quedaría definido como:

$$\min_{\alpha} \frac{1}{2}\alpha^T Q\alpha - y_i \cdot \alpha + \epsilon\alpha \;\; s.t. \;\; 0 \leq \alpha_i \leq C, \;\; \forall i = 1, ..., 2N \;, \tag{3.4.35}$$

donde $Q = x_i^T x_i$ es la matriz de núcleos y e un vector de ceros que contiene un 1 en la iésima posición. Además, resolviendo este problema se puede obtener w^* como

$$w^* = \sum_{i=1}^{N}((\alpha_i^+)^* - (\alpha_i^-)^*)x_i \;. \tag{3.4.36}$$

Una vez aquí, el método de Descenso Dual por Coordenadas usa el mismo algoritmo visto con anterioridad con la única salvedad de la definición del subproblema de una variable que resuelve en cada iteración, que ahora va a ser el siguiente.

$$\min_{d} \frac{1}{2} \bigtriangledown_{ii}^2 f(\alpha^{k,i})d^2 + \bigtriangledown_i f(\alpha^{k,i})d \quad s.t. \;\; 0 \leq \alpha^{k,i} + d \leq C \;. \tag{3.4.37}$$

Por tanto, la forma de actualizar los coeficientes α es:

$$\alpha_i \leftarrow \min\left(\max\left(\alpha_i - \frac{\bigtriangledown_i f(\alpha)}{\bigtriangledown_{ii}^2 f(\alpha)}, 0\right), C\right) \;. \tag{3.4.38}$$

Como se puede ver, es muy similar a la actualización de los coeficientes que se usaba para SVC en la fórmula (3.4.29). La diferencia se encuentra en que, como se ha dicho anteriormente, el problema SVR tiene $2N$ variables y, por tanto, van a existir dos formas de actualizar los gradientes, una por cada mitad de las variables.

$$\bigtriangledown_i f(\alpha) = \begin{cases} (Q(\alpha^+ - \alpha^-))_i + \epsilon - y_i & si \quad 1 \leq i \leq N \;; \\ -(Q(\alpha^+ - \alpha^-))_{i-N} + \epsilon - y_{i-N} & si \quad N+1 \leq i \leq 2N \;. \end{cases} \tag{3.4.39}$$

Y por tanto,

$$\bigtriangledown_{ii}^2 f(\alpha) = \begin{cases} Q_{ii} & si \quad 1 \leq i \leq N \;; \\ Q_{i-N,i-N} & si \quad N+1 \leq i \leq 2N \;. \end{cases} \tag{3.4.40}$$

Los cálculos para la implementación de este algoritmo se pueden ver en [19]. De nuevo, los desarrollos anteriores se pueden también aplicar sobre núcleos no lineales.

3.4.3. Pegasos para Regresión

El algoritmo Pegasos desarrolla un descenso por gradiente estocástico sobre el problema primal con un tamaño de paso cuidadosamente escogido. El método solo se aplica a modelos lineales y el problema a resolver es el siguiente:

$$\min_{w} \frac{\lambda}{2}||w||^2 + \frac{1}{m}\sum_{(x,y)\in S} \ell(w; (x, y)) \;, \tag{3.4.41}$$

donde

$$\ell(w; (x, y)) = \max\{0, 1 - y\langle w, x\rangle\} \;.$$

Los pasos del algoritmo son:

1. Se elige un ejemplo (x, y) de entrenamiento aleatorio. De esta forma, el problema local a resolver sería minimizar:

$$f(w; i_t) = \frac{\lambda}{2}||w||^2 + \ell(w; (x_{i_t}, y_{i_t})) \ . \tag{3.4.42}$$

2. El problema se resuelve usando el sub-gradiente:

$$\bigtriangledown_t = \lambda w_t - \mathbb{1}[y_{i_t}\langle w_t, x_{i_t}\rangle < 1]y_{i_t}x_{i_t} \ . \tag{3.4.43}$$

   donde $\mathbb{1}[y_{i_t}\langle w_t, x_{i_t}\rangle < 1]$ es la función indicadora que toma valor 1 si su argumento es verdadero y 0 si no.

3. La actualización $(w_{t+1} \leftarrow w_t - \eta_t \bigtriangledown_t)$ se realiza usando un paso de tamaño $\eta_t = 1/(\lambda t)$:

$$w_{t+1} \leftarrow (1 - \frac{1}{t})w_t + \eta_t \mathbb{1}[y_{i_t}\langle w_t, x_{i_t}\rangle < 1]y_{i_t}x_{i_t} \ . \tag{3.4.44}$$

El pseudocódigo se puede ver en [22]. El algoritmo admite dos variaciones para mejorar la eficiencia del mismo:

- Incorporación de un paso de proyección.
- Iteraciones Mini-Batch.

La primera de ellas consiste en limitar el conjunto de soluciones admisibles a una esfera de radio $\frac{1}{\sqrt{\lambda}}$. De esta forma, la actualización para proyectar w_t sobre la esfera se define como:

$$w_{t+1} \leftarrow \text{mín}\left\{1, \frac{1/\sqrt{\lambda}}{||w_{t+1}||}\right\} w_{t+1} \tag{3.4.45}$$

Por otra parte, las iteraciones Mini-Batch consisten en usar un subconjunto de los ejemplos en cada iteración en lugar de uno solo. Los ejemplos que forman parte del subconjunto se eligen aleatoriamente. Por tanto, el problema a resolver en cada iteración es:

$$\text{mín}\, f(w; A_t) = \frac{\lambda}{2}||w||^2 + \frac{1}{k}\sum_{i \in A_t} \ell(w; (x_i, y_i)) \ . \tag{3.4.46}$$

Dicho problema se resuelve usando el subgradiente:

$$\bigtriangledown_t = \lambda w_t - \frac{1}{k}\sum_{i \in A_t} \mathbb{1}[y_i\langle w_t, x_i\rangle < 1]y_i x_i \ . \tag{3.4.47}$$

De nuevo, se vuelve a realizar la actualización con un tamaño de paso $\eta_t = 1/(\lambda t)$,

$$w_{t+1} \leftarrow (1 - \frac{1}{t})w_t + \frac{\eta_t}{k}\sum_{i \in A_t} \mathbb{1}[y_i\langle w_t, x_i\rangle < 1]y_i x_i \ . \tag{3.4.48}$$

Finalmente, comentar que tanto el Descenso Dual por Coordenadas como Pegasos se mencionan para completar teoría pero en los experimentos solo se va a usar SVR.

Capítulo 4

Experimentos

4.1. Problema a resolver

Como se comentó en el Capítulo 2, la naturaleza inestable de la energía fotovoltaica hace muy difícil la gestión de las plantas solares. Por ese motivo, en este TFM se ha intentado dar una solución a esa problemática aplicando técnicas de Machine Learning que den una predicción de la producción de energía solar por horas en cada punto seleccionado de la península.

En este capítulo se va a explicar el desarrollo del proyecto en lo concerniente a los datos y técnicas utilizados para obtener resultados, así como cada uno de los experimentos realizados.

4.1.1. Explicación de los datos y su preprocesado

Lo primero que hay que tener en cuenta a la hora de trabajar con los datos que proporciona el Centro Europeo es que las horas siguen el estándar de tiempo denominado Tiempo Universal Coordinado (UTC por sus siglas en inglés). Esto quiere decir que se toma como referencia la hora local del Meridiano Primario, que es aquel cuya longitud es 0º, dada en horas y minutos en el reloj de 24 horas. En lo concerniente a este trabajo, hay que tener en cuenta que se está trabajando con coordenadas de España, donde el rango de horas es UTC+1 o UTC+2 dependiendo de si está vigente el horario de inverno o el de verano respectivamente. Por tanto, la radiación que los datos dicen pertenecer a las 14 UTC, pertenece a las 15 o 16 hora local de España.

Por otro lado, los datos obtenidos del Centro Europeo constan de un total de 56 ficheros: tres para los años 2013, 2014 y 2015 del modelo determinista, otros tres para los años 2013, 2014 y 2015 del ensemble de control y 50 para el año 2015 del resto de ensembles del modelo de ensembles. Los ficheros del modelo determinista contienen predicciones horarias para cada uno de los años seleccionados, mientras que los ficheros del modelo de ensembles contienen predicciones trihorarias, es decir, de tres horas en tres horas.

Ambos modelos proporcionan los datos de radiación de forma acumulada, es de-

cir, la radiación representada a una hora no es la radiación medida a esa hora, sino la suma de todas las radiaciones hasta esa hora dentro del mismo día. A modo de ejemplo, en el modelo determinista la radiación que aparece a las 3 UTC es la suma de la radiación a la 1 UTC, a las 2 UTC y a las 3 UTC. Como se busca predecir la producción de forma horaria, se implementó un algoritmo para desacumular estos datos.

En un principio, los datos que se descargaron del Centro Europeo tenían resolución 0.125º para el modelo determinista y resolución 0.25º para el modelo de ensembles. Ambas resoluciones implican un conjunto amplio de coordenadas, lo que provocaba que el tiempo de computación fuera inmanejable. En el primer experimento realizado con el modelo determinista en la Sección 4.2, se explica con detalle la decisión de reducir el número de coordenadas y con ello, la resolución de los datos a 0.5º.

En ese primer experimento mencionado anteriormente se usan todas las horas del día. Sin embargo, con la intención de reducir nuevamente el conjunto de datos y además obtener unos resultados más acordes a la realidad, se procedió a eliminar las horas nocturnas, donde la radiación es siempre nula. Tras esta modificación, el rango de horas en el que se realiza la predicción comprende desde las 6 UTC hasta las 21 UTC. El cambio en la distribución de los valores de la producción obtenida mediante la eliminación de las horas de noche se puede apreciar en la Figura 4.1.1 y en la Figura 4.1.2, donde la reducción de la cantidad de valores nulos es notable.

Además de los conjuntos de datos ya comentados, se necesitaban los de radiación clear-sky de los años 2013, 2014 y 2015 para realizar la interpolación explicada en la siguiente sección. Para obtenerlos se usó la librería *pvlib* para Python. En los datos descargados había una medida de la radiación clear-sky para cada hora de cada día para cada coordenada. Por esa razón, se crearon otros ficheros a partir de los originales calculando la media de radiación clear-sky sobre la península. De esta forma se tiene un único valor promedio para cada hora de un año (sin contar la noche).

Tal y como se puede apreciar en la Figura 4.1.3, los nuevos datos de clear-sky mantienen la misma distribución de los originales. Esto quiere decir que en un día, la radiación va aumentando hasta alcanzar su punto máximo en torno al mediodía y posteriomente desciende hasta ser nula una vez entrada la noche. Al comprobar que la nombrada distribución se mantenía en cada una de las estaciones del año en los nuevos datos, se procedió a dar como válida la transformación de los datos de clear-sky en un promedio peninsular.

4.1.2. Medida del error

Existen muchas formas distintas de medir el error que cometen los modelos. Sin embargo, en este proyecto se ha elegido el Mean Absolute Error (MAE) por ser el más usado dentro de la literatura relacionada con la predicción de la producción de energía fotovoltaica. El MAE se define de la siguiente forma:

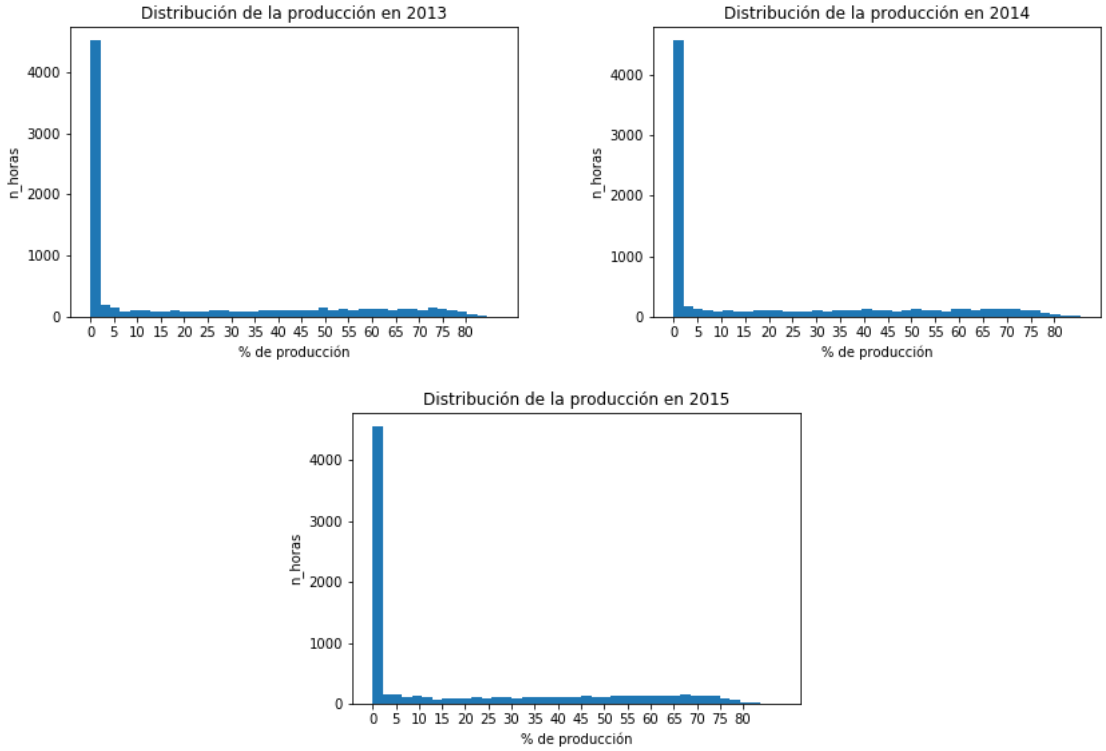


Figura 4.1.1: Distribución de la producción de energía fotovoltaica en 2013, 2014 y 2015

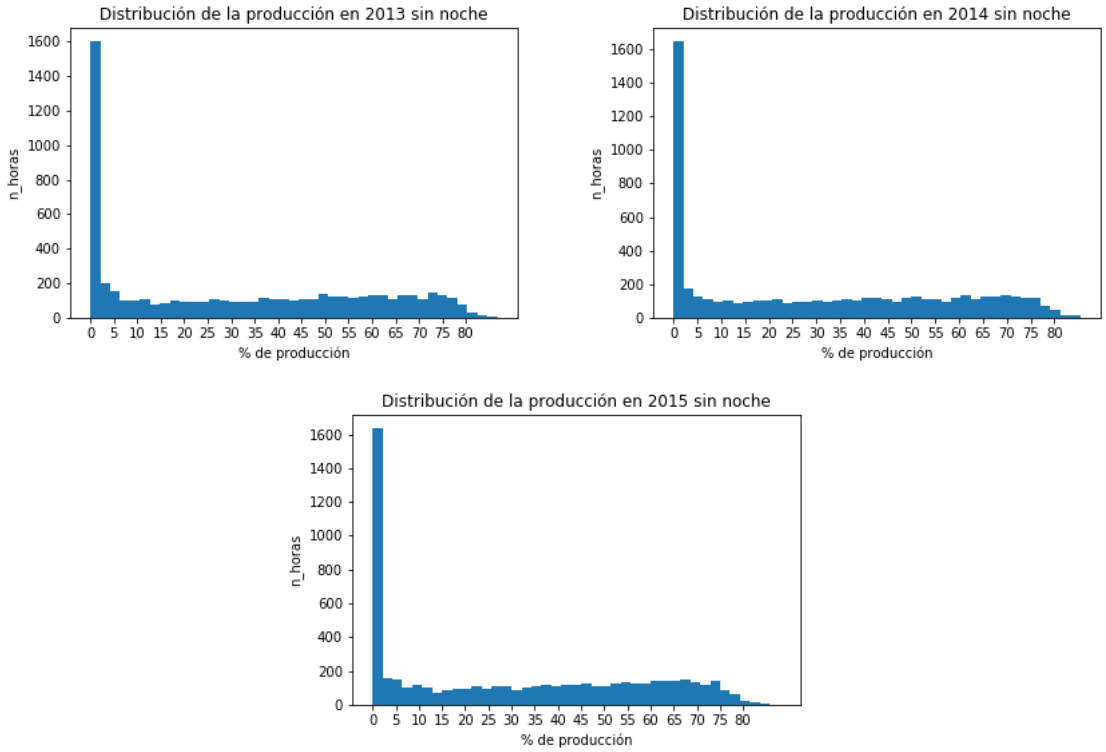
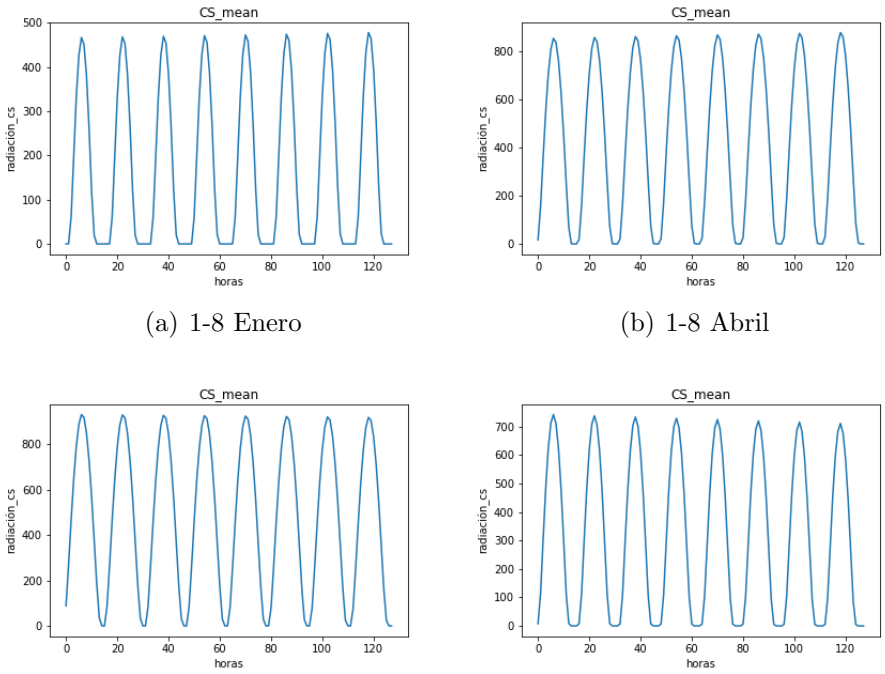


Figura 4.1.2: Distribución de la producción de energía fotovoltaica con horas reducidas en 2013, 2014 y 2015



(a) 1-8 Enero (b) 1-8 Abril

(c) 1-8 Agosto (d) 1-8 Octubre

Figura 4.1.3: Ejemplo de gráficas de la media del clear-sky en cada una de las estaciones del año.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} , \tag{4.1.1}$$

donde y_i es el valor real y $\hat{y}_i$ es el valor predicho.

Tal y como se puede apreciar en la definición, el MAE es una media de la diferencia entre los valores predichos y los valores reales. En el caso concreto de este trabajo, el error representado por el MAE es un porcentaje, ya que se está midiendo sobre la producción de energía y ésta se ha normalizado entre 0 y 100.

4.1.3. Interpolación de los datos

Tal y como se comentó anteriormente, el objetivo de este trabajo es usar el modelo de ensembles del Centro Europeo para intentar mejorar la precisión de la producción de energía fotovoltaica en cada punto de la península. Sin embargo, los datos procedentes del modelo de ensembles son trihorarios. En la Figura 4.1.4 se puede observar la diferencia entre los datos en formato horario y los datos en formato trihorario, en este caso sobre la variable clear-sky. En este contexto surge la necesidad de interpolar los valores de las horas que faltarían en la predicción.

La forma en la que se afronta el problema es la siguiente: se usan los datos trihorarios para las fases de entrenamiento, validación y finalmente test, con lo que se

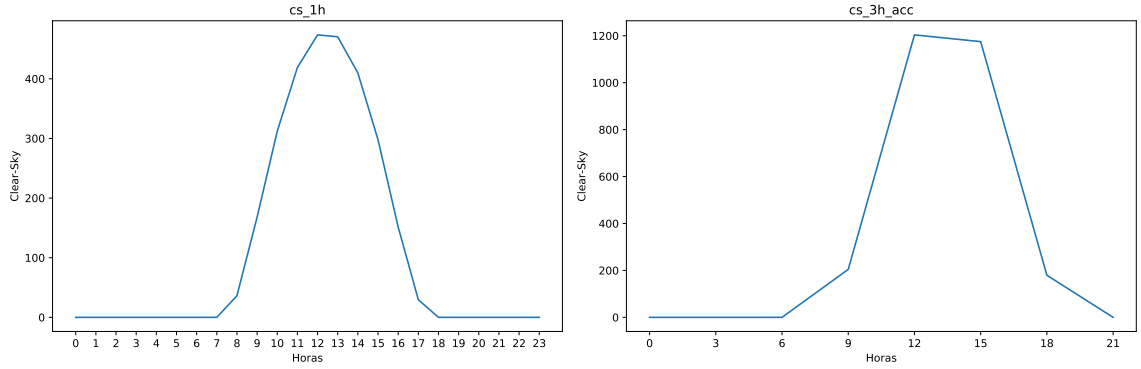


Figura 4.1.4: Gráfica de datos en formato horario vs formato trihorario.

obtendría una predicción trihoraria y posteriormente, aprovechando que la producción sigue una tendencia muy parecida a la que seguía el clear-sky, se realiza una interplación para averiguar los valores de las horas faltantes.

Algoritmo 2: Interpolación

Input:

prod ← datos de producción trihoraria
cs ← datos media clear-sky
cs_acc ← datos trihorarios de la media del clear-sky

Do:

prod_3h ← Eliminar predicciones negativas en prod.
prod_h ← extender_trihorario_a_horario (prod_3h)
cs_acc_3h ← Poner todos los valores nulos a 1 en cs_acc.
cs_acc_h ← extender_trihorario_a_horario (cs_acc_3h)
$interpol_i = \frac{cs_i}{cs\_acc_{i+1}} prod_{i+1}$

La forma de implementar la interpolación mediante clear-sky a los datos de ensembles consistiría en aplicar; por ejemplo para la hora 4:

$$\hat{p}_4 = \frac{cs_4}{cs\_acc_6} p\_acc_6 \ ; \tag{4.1.2}$$

es decir, en general se buscaría el porcentaje de radiación que correspondería a cada hora dentro de un rango y se le aplicaría a la radiación trihoraria. En el ejemplo mostrado en la ecuación (4.1.2), para obtener la producción a las 4 UTC habría que dividir la radiación estimada por clear-sky a las 4 UTC entre el total de radiación acumulada a las 6 UTC en un modelo clear-sky trihorario y multiplicarlo por la radiación trihoraria que da el modelo de ensembles para las 6 UTC.

Con el objetivo de tener una visión más general, se puede estudiar el Algoritmo 2. En él se observa que para poder realizar la operación final de interpolación, es necesario preparar los conjuntos de datos tanto de predicción de la producción como de clear-sky acumulado para que tengan todos las mismas dimensiones. Esto se

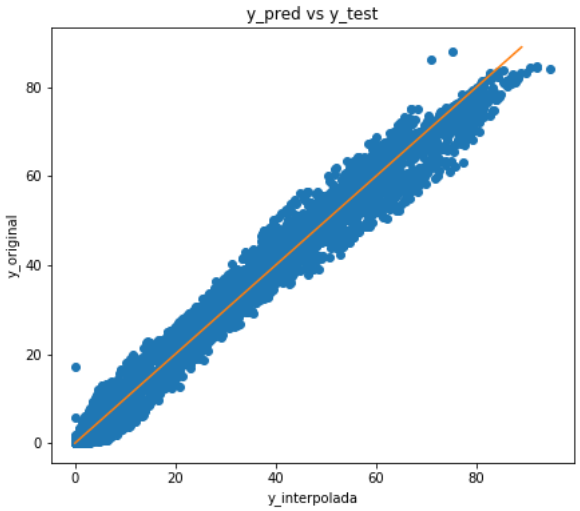


Figura 4.1.5: Comprobación de la validez de la interpolación mediante clear-sky para la producción de energía. MAE: 1.9676.

consigue mediante el método extender_trihorario_a_horario(), cuya única función es rellenar las horas faltantes en los datos trihorarios usando una copia de los datos colindantes. Por ejemplo, en el rango 3-9 UTC, se rellenarían las horas 4 y 5 UTC con el valor de las 6 UTC y las horas 7 y 8 UTC con el valor de las 9 UTC.

Para comprobar la validez de esta interpolación se usó el dataset con radiaciones clear-sky originalmente en formato horario y se creó uno nuevo con las radiaciones clear-sky en formato trihorario. Además, se usó uno de los datasets de producciones que se tenían para crear otro con producciones trihorarias. Con estos conjuntos de datos, se aplicó la interpolación explicada anteriormente para volver a conseguir el dataset horario de producciones a partir del trihorario. Tras comparar el nuevo dataset horario con el original se obtuvo un MAE de 1.9676. Este resultado se puede ver de forma gráfica en la Figura 4.1.5.

Aunque el grosor de la gráfica parezca representar un MAE entre 5 y 6, hay que tener en cuenta varios factores. El primero de ellos el grosor con el que se representa cada uno de los puntos, dando lugar a que valores con mucho error incrementen la gráfica global. Además, recordar que en este experimento la producción es nula entre las 0-6 y las 21-24, haciendo que disminuyan los errores cometidos en estos intervalos al ser valores fáciles de interpolar.

4.1.4. Modelos de Machine Learning y procedimiento para los experimentos

El modelo de machine learning que se va a usar para los experimentos de este trabajo es SVR, siendo conocida su efectividad en los problemas relacionados con la predicción de la producción de la energía fotovoltaica, tal y como explican en

[23] [2] [24] [25]. Anteriormente se ha explicado este modelo de forma teórica, así que a continuación se va a dar una breve explicación de los parámetros que necesita ajustar este modelo de una forma más ligada a la práctica. Se recuerda la fórmula del problema primal de SVR definida en (3.3.10):

$$\min_{w,w_0,\xi,\xi'} \frac{1}{2}||w||^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i') \ ;$$

$$\begin{aligned} t.q. \quad & x_p^T w + w_0 - y_i \leq \epsilon + \xi_i' \ , \\ & y_i - x_i^T w - w_0 \leq -\epsilon - \xi_i \ , \\ & \xi_i, \xi_i' \geq 0 \ , \end{aligned}$$

los parámetros a ajustar de este modelo son:

- C: es el parámetro de penalización para el error. Esto quiere decir que cuanto mayor sea su valor, menor error se permitirá en el modelo, pero a la vez, la capacidad para adaptarse a nuevos datos será menor.
- ϵ: hace referencia al tamaño del "tubo"asociado al error "ϵ-insensitive". Hace que los patrones que no sigan una especie de regresión por dentro del "tubo"definido tengan una mayor penalización.
- γ: es el parámetro del kernel gaussiano definido como $\gamma = \frac{1}{2\sigma^2}$. Se recuerda la fórmula del kernel gaussiano en (3.3.24):

$$K(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) = exp(-\gamma||x_i - x_j||^2) \ .$$

Una vez aclarados los parámetros a ajustar del modelo, se va a hacer una revisión de los conjuntos de datos que se han usado para los experimentos. Se han usado tres divisiones de los datos para las distintas fases de los modelos. Los datos pertenecientes al año 2013, independientemente de su origen, se han usado como conjunto de entrenamiento (train); los datos de 2014 como conjunto de validación y los datos de 2015 como conjunto de test. En resumen, los modelos se ajustan con los datos de 2013 y 2014 para tratar de predecir la producción total de la España peninsular en 2015.

Como se comentó anteriormente, hay que realizar una hiperparametrización de SVR para encontrar los parámetros óptimos del modelo para cada experimento. La forma de hacerlo consistió en definir unos rangos para cada uno de los parámetros y probar todas las combinaciones posibles entre ellos aplicados a los datos de entrenamiento. Para cada conjunto de valores de los tres parámetros se construía un modelo sobre los datos de 2013 y se predecía sobre 2014. Cada uno de estos resultados se almacenaba en un fichero. De esta forma, se podía saber cuál era el mejor resultado obtenido calculando el MAE de 2013 y 2014 y con qué conjunto de parámetros se había logrado. Posteriormente, esos parámetros se usaban para entrenar el modelo final sobre 2013, que predeciría sobre 2015. Finalmente, se obtiene una medida del error usando el MAE. En el primer experimento se explicará de nuevo este proceso,

orientándolo de forma más concreta al desarrollo.

Este procedimiento general es la base de todos los experimentos, aunque en algún caso presentan además otras particularidades que se comentarán en cada uno de ellos.

4.2. Predicción con el modelo determinista horario

En este primer experimento el objetivo es predecir la producción de energía fotovoltaica que se obtuvo en 2015 a partir de los datos de 2013 y 2014 usando los datos del modelo determinista en formato horario. Por tanto, como se ha explicado en la sección anterior, se van a usar tres conjuntos de datos: 2013 para entrenamiento, 2014 para validación y 2015 para test. El error se va a medir en términos del Mean Absolute Error (MAE).

El proceso que se ha seguido para este experimento es el siguiente: En primer lugar se ha ejecutado el proceso de validación, en el que se han entrenado tantos modelos como combinaciones de parámetros se tenían (en este caso 120) y se ha usado la matriz de 2014 para calcular el MAE de validación. Posteriormente, se han escogido los parámetros que han obtenido un menor MAE en validación y se han usado para entrenar un modelo cuyo conjunto de test va a ser la matriz de datos de 2015. Tras este paso, se obtiene el MAE en test, que es realmente el que interesa.

El rango de los parámetros que se ha usado para este experimento se ha definido de la siguiente forma:

```
1        C = [10.**k for k in range (0,5)]
2        gamma = list(np.array([2.**k for k in range(-2, 4)])/n_dimensiones
3        epsilon = list(y_train.std() * np.array([2.**k for k in range(-6,-2)]))
```

Como se puede ver, el parámetro C va a tener siempre los mismos valores, mientras que γ depende del número de dimensiones que se tengan y ϵ de la desviación estándar de los datos de producción de 2013. Esta fórmula se ha usado como base para todos los experimentos, aunque en algunos casos se ha variado el rango de los valores que pueden tomar los parámetros con el objetivo de obtener mejores resultados.

Para la predicción con el modelo determinista horario se llevaron a cabo dos experimentos: uno con los datos a resolución 0.125 y otro con los datos reducidos a resolución 0.5. Como estos experimentos forman la base de este trabajo, sirvieron para enfocar el resto de experimentos que les siguieron. Por ese motivo en ambos experimentos se usaron todas las horas del día en lugar de quitar las horas correspondientes a la noche, ya que todavía se estaban estudiando las formas de reducir la dimensión del problema.

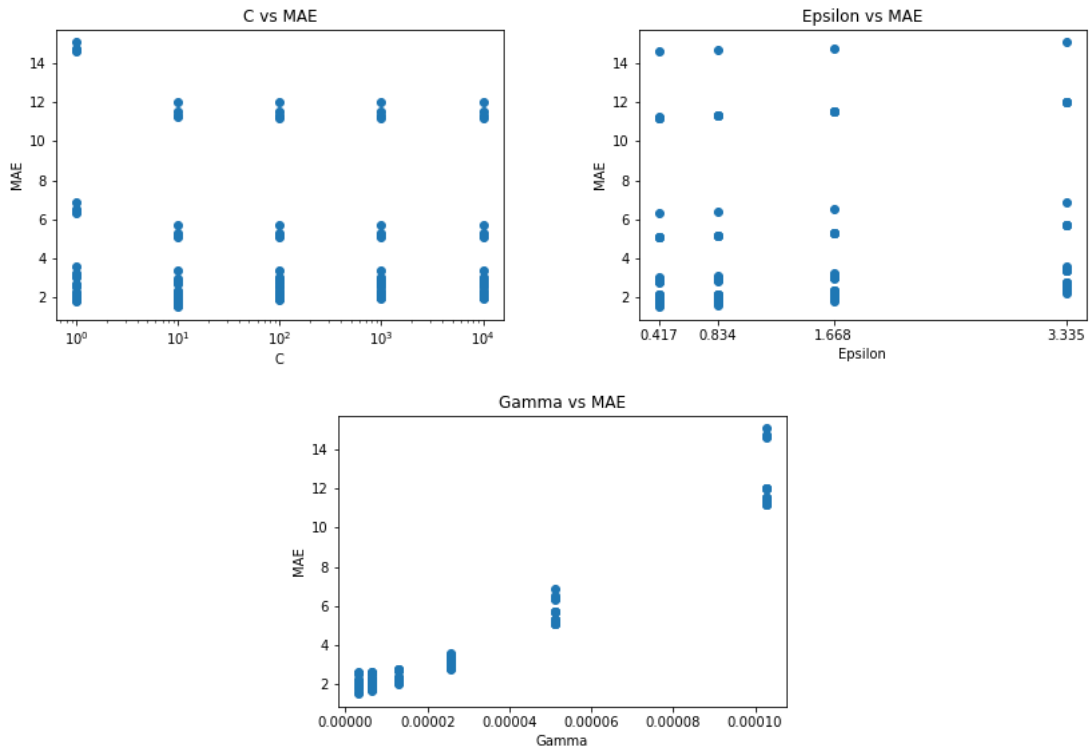


Figura 4.2.1: Valores de los parámetros con el MAE obtenido para cada uno para el modelo determinista horario a resolución 0.125º.

4.2.1. SVR determinista con resolución 0.125º

En este experimento, el conjunto de datos de entrenamiento cuenta con 8760 filas y 77970 columnas. De forma particular para esta primera prueba, los posibles valores para cada parámetro con los siguientes:

- C = 1.0, 10.0, 100.0, 1000.0, 10000.0
- ϵ = 0.4169, 0.8338, 1.6675, 3.3351
- γ = 3.2064e-06, 6.4127e-06, 1.2825e-05, 2.5651e-05, 5.1302e-05, 0.0001

Tras lanzar el proceso de validación sobre los datos de 2014 los parámetros que obtenían un menor error eran **(C = 10.0, ϵ : 0.4169, γ : 3.2064e-06)**, con un **MAE de 1.5375**. Además, se pudo comprobar la influencia de cada parámetro en el error obtenido, tal y como se puede observar en la Figura 4.2.1.

Si se analizan las gráficas de esta figura, se puede observar que el rango escogido para el parámetro C es adecuado, ya que el valor escogido es $C = 10$ y tanto valores mayores como menores obtienen un MAE mayor. Sin embargo, tanto ϵ como γ obtienen el menor MAE con su valor más pequeño, por lo que sería necesario ampliar los valores de los rangos de estos dos parámetros para obtener mejores resultados. Sin embargo, como este experimento se está realizando principalmente con el objetivo de comparar los resultados obtenidos bajo distintas resoluciones de los datos, no va a

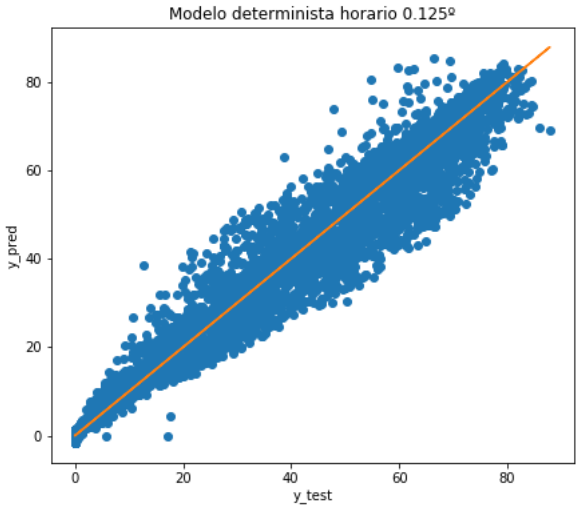


Figura 4.2.2: Valores obtenidos de la predicción frente a los valores reales con el modelo determinista a resolución 0.125º (MAE = 2.4199).

ser necesario calcular los hiperparámetros óptimos sino que va a bastar con comparar ambos experimentos realizados bajo las mismas condiciones.

Tras comprobar que la hiperparametrización elegida como óptima se ajusta a lo que se quiere probar, se ejecuta el modelo entrenado con estos parámetros para comprobar el error que se obtiene con los datos de test. En este caso se obtiene un **MAE de 2.4199**. Este valor tiene sentido, ya que debe ser un poco mayor que el de validación debido a que se le están mostrando al modelo datos a los que no ha tenido oportunidad de ajustarse.

Por otro lado, en la Figura 4.2.4 se comparan los valores de producción obtenidos frente a los reales medidos en 2015. La gráfica tiene una tendencia lineal diagonal, lo que demuestra que el error obtenido en este experimento, pese a que la hiperparametrización se podría mejorar, es mínimo. Sin embargo, como pasaba en la sección anterior, el grosor de la gráfica puede producir confusión si no se tienen en cuenta los factores que se comentaron. Además esta gráfica se ve más afectada debido a la gran cantidad de datos que intenta representar.

4.2.2. SVR determinista con resolución 0.5

En este segundo experimento, la resolución de los datos es menor y, por tanto, las matrices quedan reducidas a 5840 filas y 4176 columnas, ya que además de reducir el número de coordenadas se han eliminado las horas de noche. Para poder comparar los resultados obtenidos con los del experimento anterior, se van a usar los mismos valores para el rango de los parámetros.

En la fase de validación se obtuvieron como los mejores parámetros **(C = 10.0,** ϵ

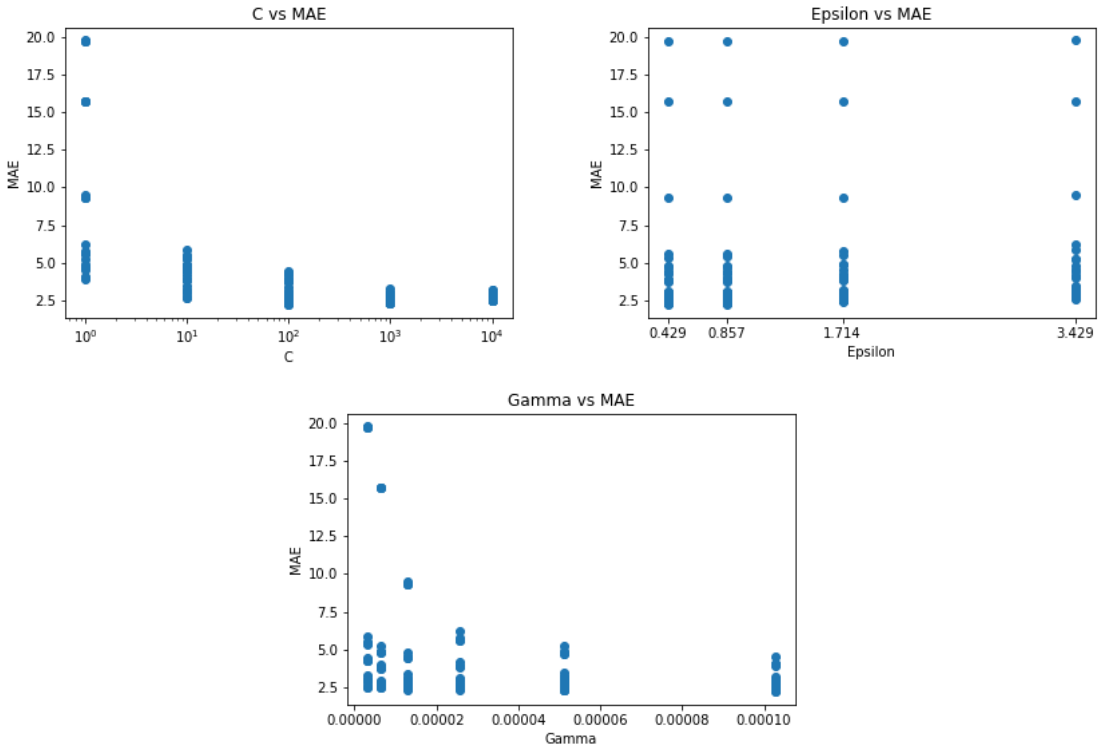


Figura 4.2.3: Valores de los parámetros con el MAE obtenido para cada uno para el modelo determinista horario a resolución 0.5º.

$= 0.4286$, $\gamma = 0.0010$), con un **MAE de 2.2082**. De nuevo, si prestamos atención a la Figura 4.2.3 se puede observar que, como ocurría en el experimento anterior, los parámetros ϵ y γ no están bien ajustados. Sin embargo, esto permite comparar los resultados con los del experimento anterior.

En la Tabla 4.2.1 se muestran las diez primeras combinaciones de parámetros que obtuvieron el error más pequeño. Como se puede ver, el parámetro C es fijo para las 8 primeras, mientras que tanto ϵ como γ son más variables. Esto hace ver que estos dos últimos parámetros son más influyentes en el resultado final.

Finalmente, en la fase de test se obtiene un **MAE de 2.4459**. Si se compara este valor con el obtenido en el experimento anterior, en el que los datos tenían resolución 0.125º se observa que se ha obtenido un error similar. De esta forma se llega a la conclusión de que mantener la cantidad inicial de datos no mejora de forma determinante el modelo, ya que se han obtenido resultados similares con una reducción significativa de los mismos. En la Figura 4.2.4 se muestra de forma gráfica el error de test obtenido en este experimento. En esta ocasión puede parecer que la gráfica es mucho mejor que la del experimento anterior cuando el MAE es ligeramente peor. Esto es así debido a que antes se tenían las horas de noche con valor nulo y provocaban que hubiera muchos aciertos en ese rango debido a que la radiación nula es fácil de predecir.

Una vez comprobada la validez de la reducción de dimensiones de los datos, se

C	ϵ	γ	MAE
10.0	0.428600	0.000479	2.145441
100.0	0.428600	0.000239	2.147600
10.0	0.857201	0.000958	2.165136
100.0	0.428600	0.000120	2.188276
100.0	0.857201	0.000239	2.195690
10.0	0.857201	0.000479	2.201464
100.0	0.428600	0.000479	2.203042
10.0	0.428600	0.001916	2.215317
100.0	0.857201	0.000120	2.236431
100.0	0.857201	0.000479	2.259804

Cuadro 4.2.1: Las diez combinaciones de parámetros con menor MAE para el modelo determinista horario a resolución 0.5º.

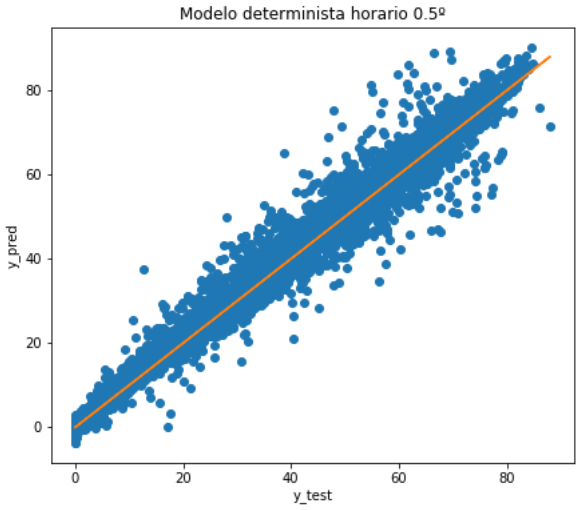


Figura 4.2.4: Valores obtenidos de la predicción frente a los valores reales con el modelo determinista horario a resolución 0.5º (MAE = 2.4459).

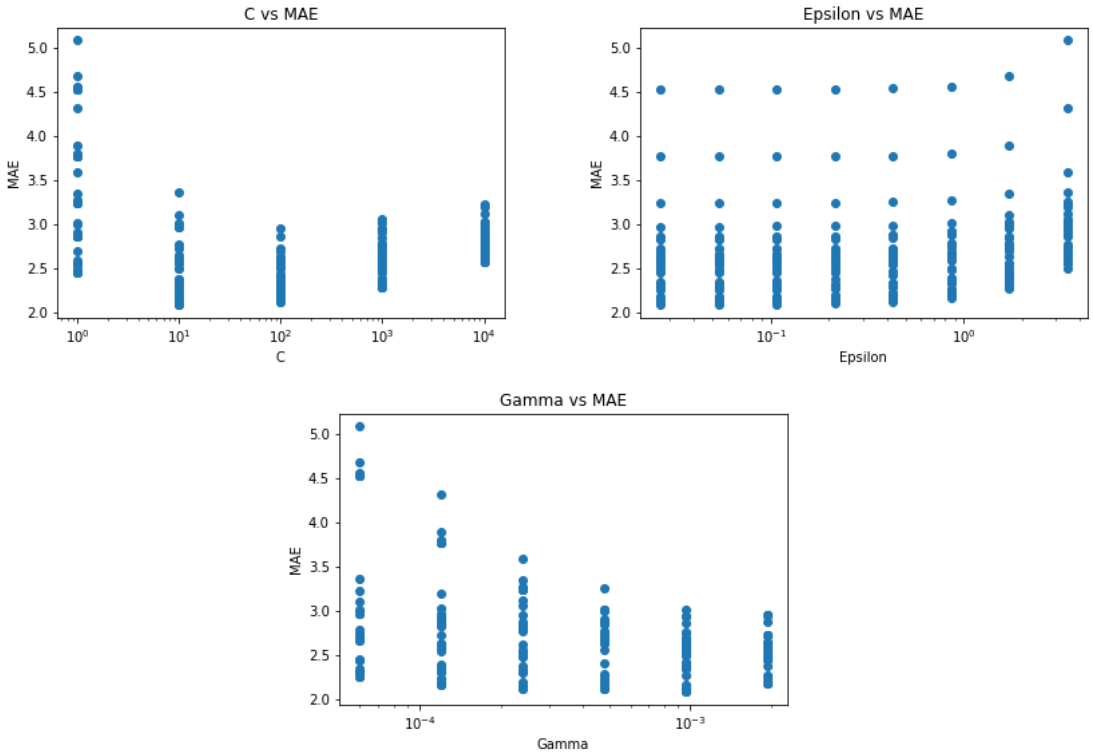


Figura 4.2.5: Valores de los parámetros con el MAE obtenido para cada uno para el modelo determinista horario a resolución 0.5º tras ampliar el rango de valores de los parámetros.

procede a repetir la hiperparametrización de este mismo experimento con el objetivo de obtener los parámetros ϵ y γ óptimos. En este caso, la cantidad de valores posibles para los parámetros se ha ampliado hasta 240. En consecuencia, los parámetros óptimos son $\mathbf{(C = 10.0,\ \epsilon = 0.02679,\ \gamma = 0.0010)}$, con un MAE en validación de **2.0838**.

A diferencia de la otra vez, en la Figura 4.2.5 se puede observar que el valor que ha tomado el parámetro γ sí es el óptimo, mientras que el parámetro ϵ sigue manteniendo el menor valor que se le ha proporcionado como opción. Se podría volver a hiperparametrizar con un rango mayor de valores para este parámetro, pero sería un esfuerzo considerable en tiempo comparado con la mejora que supondría en el error del experimento. Esto también queda reflejado en la Tabla 4.2.2, donde se observa que para un mismo valor de C y γ los valores de ϵ toman diferentes valores sin apenas afectar al MAE obtenido.

Finalmente, el error en la fase de test del modelo es de **2.2662** y en la Figura 4.2.6 se muestran los valores de predicción de la producción obtenidos mediante el modelo comparados con los valores reales de producción para 2015. Como se puede observar, se sigue una tendencia lineal en diagonal siguiendo el ideal representado por la línea naranja.

	C	ϵ	γ	MAE
1	10.0	0.026788	0.000958	2.083756
2	10.0	0.053575	0.000958	2.084785
3	10.0	0.107150	0.000958	2.088773
4	10.0	0.214300	0.000958	2.098331
5	10.0	0.428600	0.000958	2.117906
6	10.0	0.053575	0.000479	2.118713
7	10.0	0.107150	0.000479	2.119196
8	10.0	0.026788	0.000479	2.119443
9	100.0	0.053575	0.000239	2.122215
10	100.0	0.026788	0.000239	2.122296

Cuadro 4.2.2: Las diez combinaciones de parámetros con menor MAE para la repetición de la hiperparametrización del modelo determinista horario a resolución 0.5º.

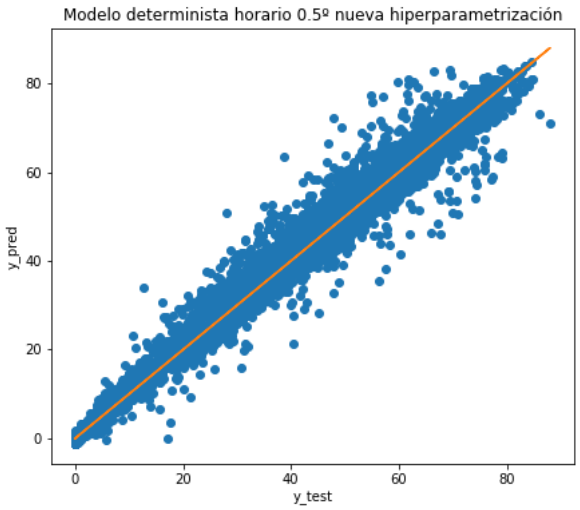


Figura 4.2.6: Valores obtenidos de la predicción frente a los valores reales para el modelo determinista horario a resolución 0.5º tras la repetición de la hiperparametrización (MAE = 2.2662).

4.3. Predicción con el modelo determinista trihorario

Tal y como se comentó anteriormente, los datos de los ensembles se encuentran en formato trihorario, por lo que es necesario usar una interpolación mediante clear-sky para predecir en formato horario. Por tanto, el objetivo principal de este experimento es comprobar que dicha interpolación obtiene buenos resultados interpolando resultados en el modelo determinista para el cual ya se tienen resultados y de esta forma poder aplicarla en los experimentos con ensembles.

En primer lugar se transformaron las matrices que contenían los datos del modelo determinista con las que se estuvo trabajando en los apartados anteriores al formato trihorario. A continuación, se realizaron los procedimientos de entrenamiento, validación y test en este mismo formato de los que se obtuvieron los primeros errores de validación y test. Finalmente, se aplicó la interpolación mediante clear-sky a la producción obtenida y se calculó el error cometido, comparando el error resultante con el obtenido para los datos horarios.

El conjunto de datos determinista convertido al formato trihorario tiene unas dimensiones de 2920 x 4176. Los posibles valores de los parámetros utilizados en la hiperparametrización son los siguientes:

- C = 1.0, 10.0, 100.0, 1000.0, 10000.0
- ϵ = 0.0774, 0.1547, 0.3095, 0.6189, 1.2379, 2.4757, 4.9514, 9.9029
- γ = 5.9866e-05, 0.0001197, 0.0002395,0.0004789, 0.0009579, 0.001916

En esta ocasión también hizo falta ampliar el número de posibles combinaciones de parámetros para obtener los hiperparámetros óptimos, que tal y como se obtuvo en la fase de validación son **(C = 100.0, ϵ = 0.07737, γ = 0.0009579)**, con un **MAE de 5.4631**. Las conclusiones respecto a cada uno de los parámetros se repiten con las del experimento anterior, quedando reflejadas en la Figura 4.3.1 y en la Tabla 4.3.1.

Por otra parte, el MAE obtenido en la fase de test es de **6.0932**. Este valor es aceptable, ya que al estar prediciendo en formato trihorario el error va a ser aproximadamente el triple del que se obtuvo en el horario, debido a que, dicho de forma muy trivial, cada hora aglutina el error de tres horas diferentes. Nuevamente, en la Figura 4.3.2 se muestra el error de test. Tal y como se puede apreciar en la gráfica, el eje y en vez de llegar hasta 100 como antes llega hasta 250. Esto es debido a que ahora se está trabajando con datos trihorarios, lo que quiere decir que, repitiendo lo que se dijo antes, cada punto tendra la suma de 3 horas de producción y por tanto, el máximo que se puede obtener es aproximadamente 240.

Finalmente, la parte más interesante de este experimento consiste en aplicar el algoritmo de interpolación explicado en la Sección 4.1.3 a las predicciones trihorarias

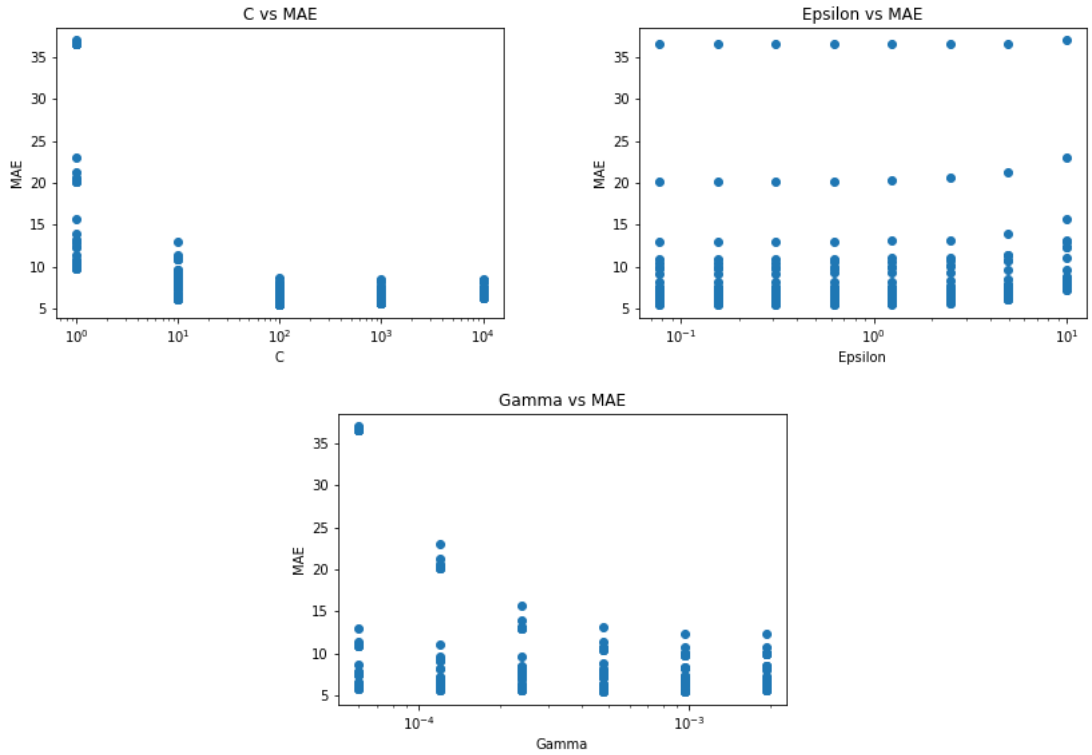


Figura 4.3.1: Valores de los parámetros con el MAE obtenido para cada uno para el modelo determinista trihorario.

	C	ϵ	γ	MAE
1	100.0	0.077366	0.000958	5.463069
2	100.0	0.154733	0.000958	5.466096
3	100.0	0.309465	0.000958	5.474051
4	100.0	0.618930	0.000958	5.487283
5	100.0	0.077366	0.000479	5.505756
6	100.0	0.154733	0.000479	5.507156
7	100.0	0.309465	0.000479	5.518332
8	100.0	1.237860	0.000958	5.527919
9	100.0	0.618930	0.000479	5.549639
10	100.0	1.237860	0.000479	5.618047

Cuadro 4.3.1: Las diez combinaciones de parámetros con menor MAE tras probar las 240 combinaciones de parámetros para el modelo determinista trihorario.

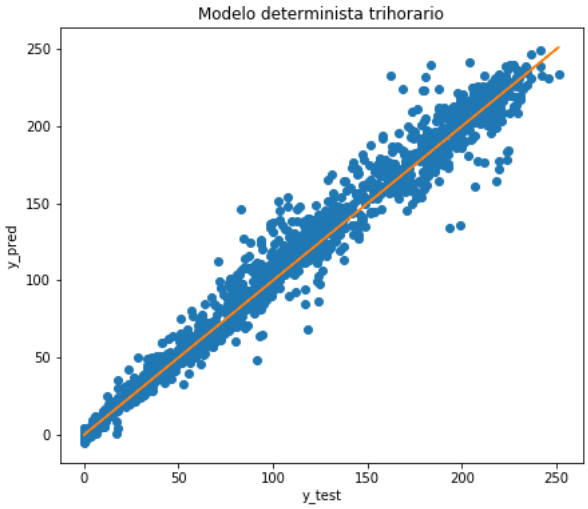


Figura 4.3.2: Valores obtenidos de la predicción frente a los valores reales con el modelo determinista trihorario (MAE = 6.0932).

para comparar el error obtenido con el que se conoce de experimentos anteriores. En este caso se ha obtenido un **MAE de 3.2487**, que al compararlo con el 2.2662 que se obtuvo en el experimento en formato horario se puede ver que la interpolación incrementa el error en aproximadamente 1. La comparación entre la producción interpolada y la real se puede ver en la Figura 4.3.3.

Este error de interpolación se ha considerado razonable y, por tanto, se da por válido el algoritmo de interpolación para su uso en experimentos con datos trihorarios como son los de los ensembles.

4.4. Predicción con Ensembles Meteorológicos

En esta sección se describe el experimento para el que se han estado haciendo todas las comprobaciones previas. Como el modelo de ensembles del Centro Europeo se compone del ensemble de control y de las 50 variaciones, se van a realizar dos experimentos. En el primero de ellos se usarán los datos de 2013, 2014 y 2015 del modelo de ensembles tal y como se ha venido haciendo hasta ahora. Con esta prueba se quiere comprobar que el resultado es similar al obtenido con el modelo determinista, ya que según la documentación, este ensemble es el más parecido al modelo determinista.

Como segundo experimento se usarán los hiperparámetros obtenidos en la validación con el ensemble de control para predecir sobre el año 2015 de cada uno de los 50 ensembles restantes. De esta forma se van a obtener 50 predicciones distintas, que se unificarán mediante una media y una mediana.

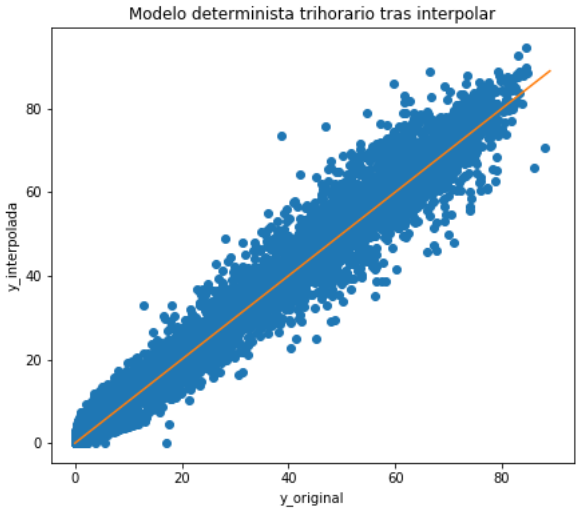


Figura 4.3.3: Resultado de interpolar frente al resultado original con el modelo determinista trihorario (MAE = 3.2487).

4.4.1. Experimento con el ensemble de control

Para este experimento se siguieron las mismas pautas de los experimentos anteriores, siendo necesario también aquí usar 240 combinaciones de parámetros para lograr que dos de los tres hiperparámetros tomaran su valor óptimo. El rango de los valores de los parámetros que se usaron se define a continuación:

- C = 1.0, 10.0, 100.0, 1000.0, 10000.0
- ϵ = 0.0270, 0.0539, 0.1078, 0.2157, 0.4314, 0.8627, 1.7254, 3.4508
- γ = 5.9866e-05, 0.0001197,0.0002395,0.0004789,0.0009579,0.001916

En la fase de validación se obtuvo que la combinación óptima de valores de los hiperparámetros era **(C = 100.0, ϵ = 0.1547, γ = 0.0009579)**, que como se puede comprobar es la misma que la obtenida para el modelo determinista en formato trihorario a excepción del parámetro ϵ. En validación el MAE alcanzó un **5.0914**, similar al obtenido con el modelo determinista trihorario. Nuevamente, se puede comprobar el comportamiento de cada parámetro en la Figura 4.4.1 y en la Tabla 4.4.1.

Finalmente, en la fase de test se obtuvo un **MAE de 5.3997**, un poco menor al obtenido en la misma fase del experimento con el modelo determinista trihorario. En la Figura 4.4.2 se puede apreciar la comparación entre los valores predichos y los valores reales de la producción en formato trihorario.

Lo que se hizo a continuación fue usar el algoritmo de interpolación mediante clear-sky para poder hacer una estimación de la predicción para las horas que no aparecen en el formato trihorario. Tras la ejecución se obtuvo un **MAE de 3.0049**. En la Figura 4.4.3 se puede observar este resultado de forma más gráfica. Al comparar

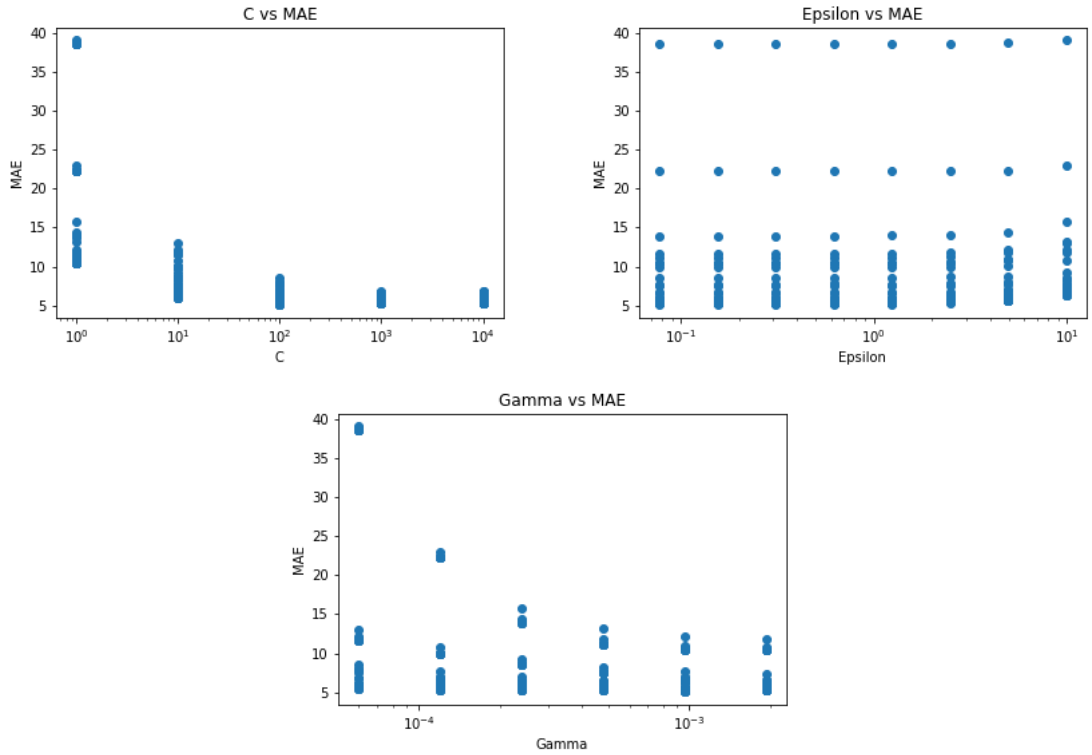


Figura 4.4.1: Valores de los parámetros con el MAE obtenido para cada uno para el modelo del ensemble de control.

	C	ϵ	γ	MAE
1	100.0	0.154733	0.000958	5.091352
2	100.0	0.077366	0.000958	5.091540
3	100.0	0.309465	0.000958	5.094268
4	100.0	0.618930	0.000958	5.104518
5	100.0	1.237860	0.000958	5.158648
6	1000.0	0.154733	0.000120	5.218653
7	1000.0	0.077366	0.000120	5.218817
8	1000.0	0.309465	0.000120	5.221666
9	1000.0	0.618930	0.000120	5.238782
10	100.0	0.077366	0.001916	5.251637

Cuadro 4.4.1: Las diez combinaciones de parámetros con menor MAE para el modelo del ensemble de control.

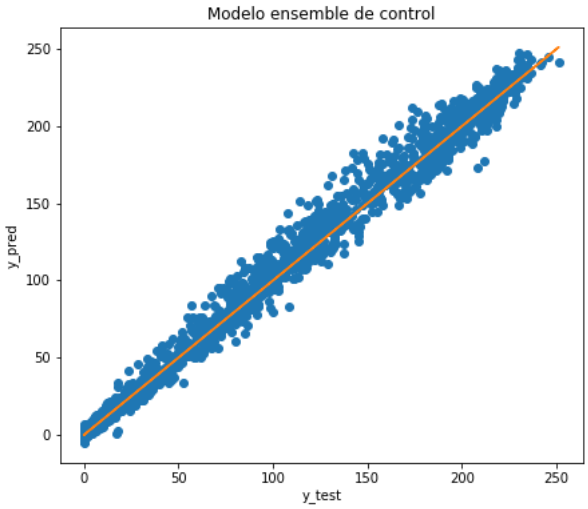


Figura 4.4.2: Valores obtenidos de la predicción frente a los valores reales con el modelo del ensemble de control (MAE = 5.3997).

este error con el conseguido por el modelo determinista trihorario tras la interpolación, se puede observar que es un poco menor, pero no demasiado. Por tanto, la conclusión que se saca de este experimento es que, efectivamente, el ensemble de control es muy parecido al modelo determinista y usarlo en su lugar no proporciona grandes beneficios.

4.4.2. Ensembles no control

Tal y como se comentaba en la sección anterior, en este experimento se han usado los 50 ensembles que introducen variaciones a la predicción determinista como 50 conjuntos de test diferentes. Para la parte de entrenamiento y validación se han usado los conjuntos de datos de 2013 y 2014 del ensemble de control. Con los mejores parámetros obtenidos se procedió a entrenar con los datos de 2013 del ensemble de control y a predecir con cada uno de los ensembles.

En total se obtuvieron 50 predicciones distintas, pero muy parecidas entre ellas, como se muestra en la Tabla 4.4.2. Por simplicidad se van a mostrar únicamente las tablas de cuatro ensembles elegidos de forma aleatoria en la Figura 4.4.5.Al tratar los resultados como una combinación de todos ellos se obtiene de **media un MAE de 5.7767 y una mediana de 5.7874**.

Como se ha venido haciendo hasta ahora, hay que aplicar el procedimiento de interpolación sobre las producciones predichas para poder conseguir la predicción horaria de la producción en el año 2015. Al haberse obtenido 50 predicciones distintas, primero se interpolará cada una de ellas y se calculará el error individual y posteriormente se volverá a calcular la media y la mediana para obtener un error global. En la Tabla 4.4.3 se muestra el error para cada uno de los ensembles tras la interpolación.

Ensemble	MAE	Ensemble	MAE
1	5.8401	26	5.7954
2	5.6191	27	5.8238
3	5.6474	28	5.7120
4	6.0125	29	5.7414
5	5.6754	30	5.8196
6	5.7080	31	5.7515
7	5.6812	32	5.8382
8	5.7559	33	5.6988
9	5.7868	34	5.8733
10	5.6382	35	5.9150
11	5.8000	36	5.7613
12	5.8132	37	5.8862
13	5.9553	38	5.8107
14	5.7977	39	5.6046
15	5.7708	40	5.8266
16	5.7364	41	5.6764
17	5.5554	42	5.7259
18	5.9783	43	5.8152
19	5.8098	44	5.8138
20	5.7915	45	5.7241
21	5.5770	46	5.7457
22	5.7923	47	5.7652
23	5.9123	48	5.9315
24	5.7880	49	5.8209
25	5.7790	50	5.7379

Cuadro 4.4.2: Errores para cada uno de los 50 ensembles en formato trihorario.

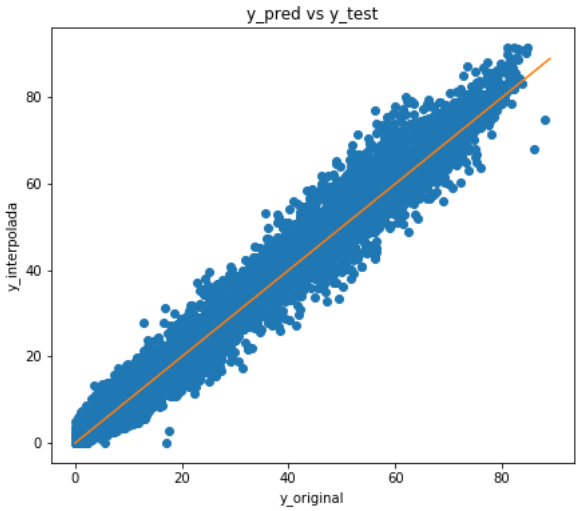


Figura 4.4.3: Resultado de interpolar frente al resultado original para el modelo del ensemble de control (MAE = 3.0049).

Como se hizo para los resultados en trihorario, se calculó la media y la mediana del MAE obtenido para agrupar todas las predicciones. En este caso la **media del MAE es de 3.1054 y la mediana de 3.1093**.

Tras este último experimento se puede concluir que los resultados de las predicciones no se ven afectados en demasía por el uso del modelo de ensembles meteorológicos.

4.5. Resultados

En esta sección se va a hacer un resumen de los resultados obtenidos en todos los experimentos y las conclusiones finales que se han extraído de los mismos.

Como se puede apreciar en la Tabla 4.5.1, el modelo determinista horario es el que mejores resultados obtiene. Sin embargo, como el objetivo de este trabajo es trabajar con ensembles meteorológicos los resultados obtenidos por los experimentos que usan datos en formato trihorario son los que se van a comparar entre sí. Al contrastar el modelo determinista en formato trihorario con el ensemble de control se observa que éste último mejora el error de forma considerable. Por otro lado, al comparar estos resultados con los producidos por el resto de ensembles, se ve claramente que este modelo que usa los 50 ensembles no mejora la predicción, sino que la empeora. Como conclusión general, las 50 variaciones que proporciona el Centro Europeo sobre el modelo determinista no hacen que la predicción sea más ajustada a la realidad aunque sí podría servir para proporcionar estimaciones de incertidumbre de las predicciones. Sin embargo, usar el ensemble de control sí que proporciona notables mejoras en el error obtenido.

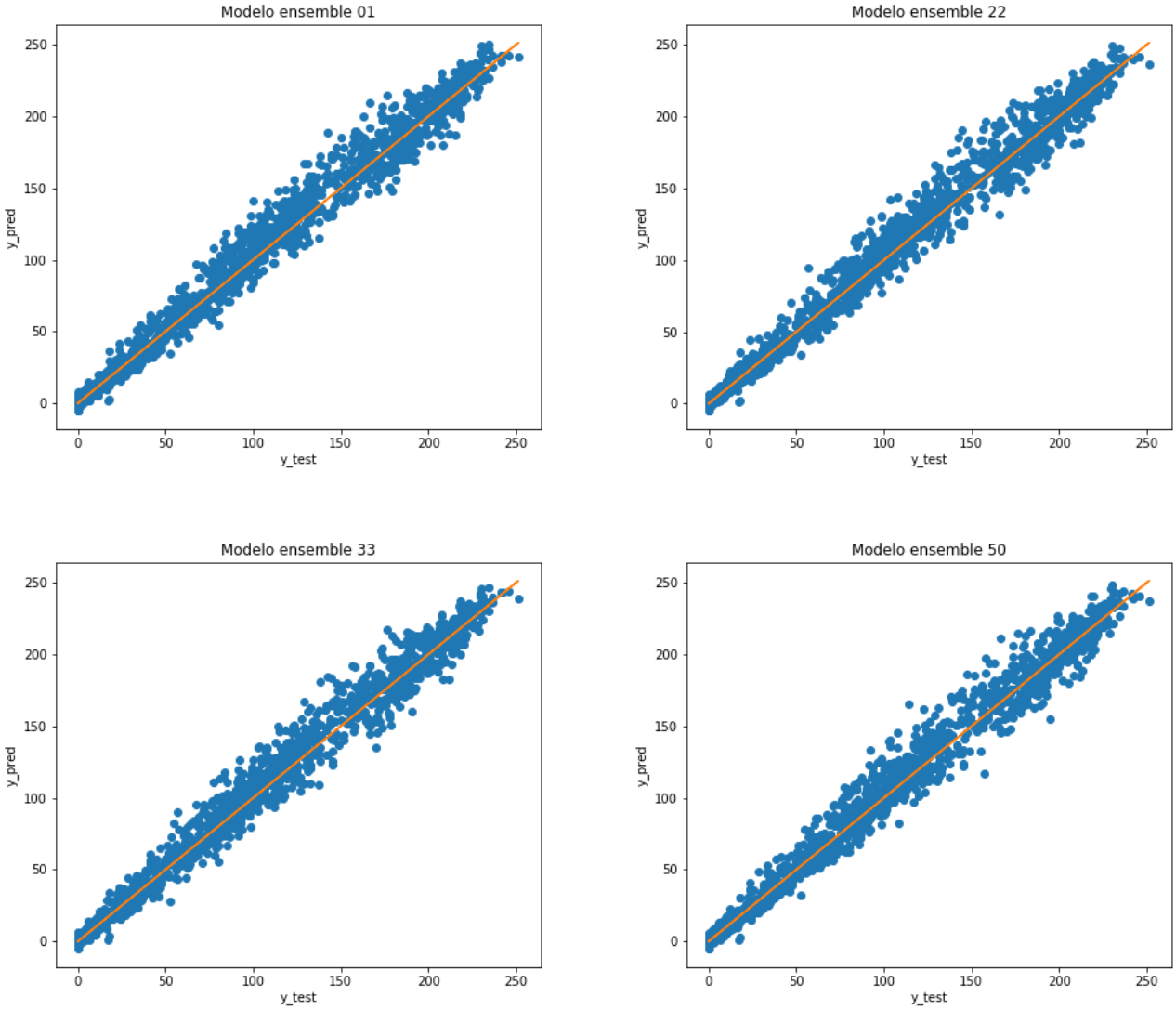


Figura 4.4.4: Plot de resultados para los ensembles trihorarios 1, 22, 33 y 50 de izquierda a derecha y de arriba a abajo.

Ensemble	MAE	Ensemble	MAE
1	3.1373	26	3.1191
2	3.0521	27	3.1261
3	3.0693	28	3.0838
4	3.1701	29	3.0928
5	3.0758	30	3.1201
6	3.0967	31	3.0847
7	3.0697	32	3.1318
8	3.1059	33	3.0806
9	3.1091	34	3.1320
10	3.0693	35	3.1503
11	3.1170	36	3.1154
12	3.1163	37	3.1179
13	3.1436	38	3.1189
14	3.1097	39	3.0486
15	3.0878	40	3.1318
16	3.0903	41	3.0867
17	3.0308	42	3.0998
18	3.1603	43	3.1230
19	3.1080	44	3.1088
20	3.1117	45	3.0876
21	3.0336	46	3.0848
22	3.1157	47	3.1086
23	3.1486	48	3.1521
24	3.1094	49	3.1232
25	3.1096	50	3.1007

Cuadro 4.4.3: Errores para cada uno de los 50 ensembles tras la interpolación.

Experimento	MAE test	MAE test interpolación
Determinista horario 0.125º	2.4199	X
Determinista horario 0.5º	2.4459	X
Determinista horario 0.5º segunda hiperparametrización	2.2662	X
Determinista trihorario	6.0932	3.2487
Ensemble de control	5.3997	3.0049
50 Ensembles (media)	5.7767	3.1054
50 Ensembles (mediana)	5.7874	3.1093

Cuadro 4.5.1: Resultados globales.

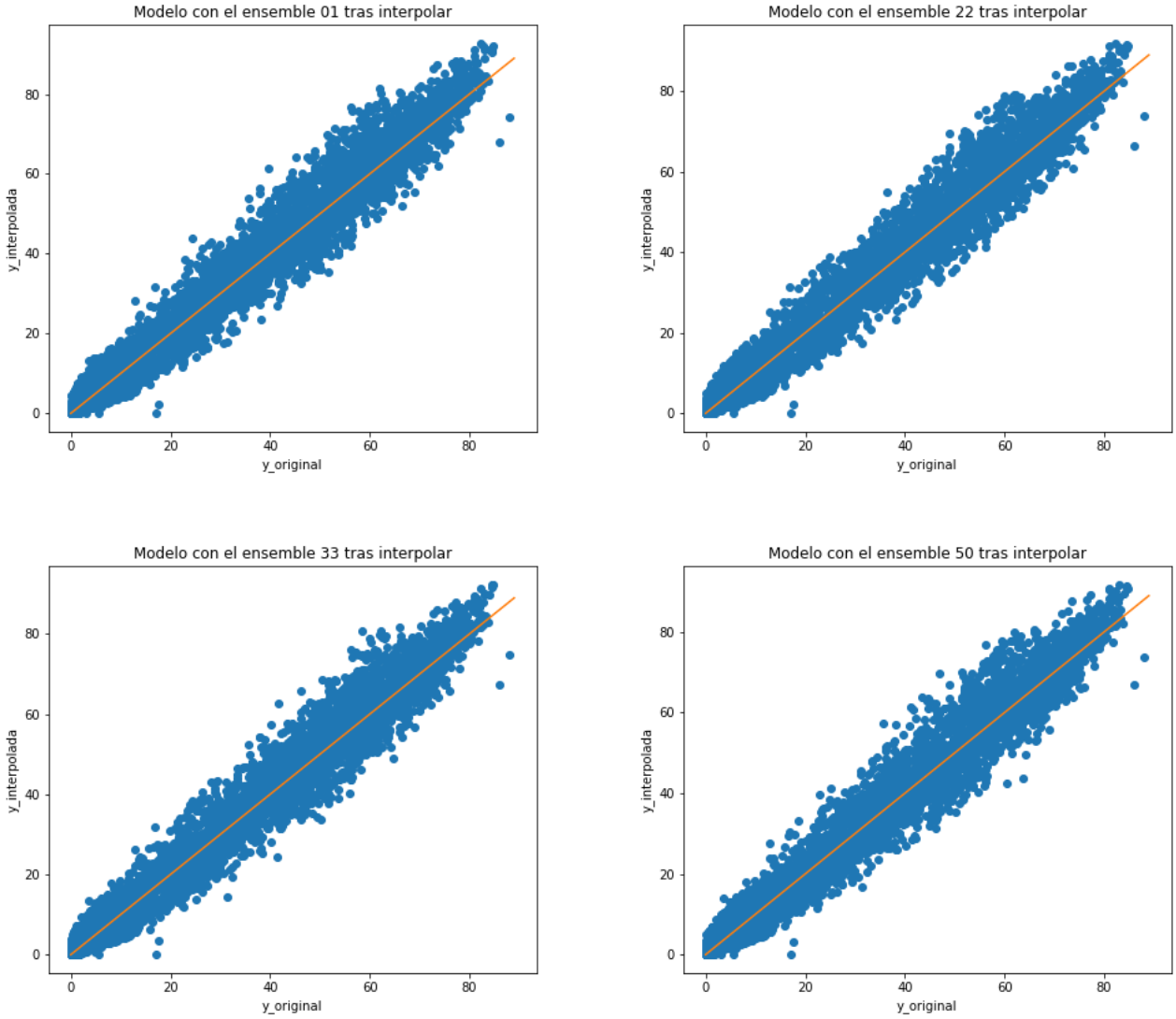


Figura 4.4.5: Plot de resultados tras la interpolación para los ensembles 1, 22, 33 y 50 de izquierda a derecha y de arriba a abajo.

Capítulo 5

Conclusión y Trabajos Futuros

5.1. Resumen

A continuación, se muestra un resumen de esta memoria.

En el Capítulo 2 se presentaron los conceptos básicos de la energía fotovoltaica así como sobre sus posibles formas de predicción, entre ellas las NWP, y el modelo clear-sky. El capítulo termina explicando la selección de variables que se llevó a cabo de cara a los modelos.

En el Capítulo 3 es donde se ahonda en los principales métodos de Machine Learning que se estudiaron para abordar el problema de predicción. Se han presentado los fundamentos de optimización convexa, esto es, los problemas primal y dual y la teoría Lagrangiana. También se ha descrito SVC y en mayor detalle SVR . Además, se han comentado algunas de las implementaciones de SVR, como son : SMO, Descenso Dual por Coordenadas y Pegasos.

En el Capítulo 4 se explican en detalle todos los experimentos realizados para este proyecto y los resultados obtenidos en cada uno. Además, se justifica el uso del algoritmo de interpolación a la hora de trabajar con datos en formato trihorario. El capítulo termina resumiendo en una tabla las conclusiones sobre los resultados obtenidos.

5.2. Interpolación por Clear-Sky

Como se ha visto en el Capítulo 4, fue necesario implementar un algoritmo de interpolación, ya que el Centro Europeo proporciona los datos en formato trihorario acumulado y para la predicción tienen que estar en horario desacumulado.

Se eligió el modelo clear-sky como referencia por su comportamiento parecido al de la producción a lo largo de un día. Sin embargo, podría ser más efectivo usar el algoritmo para poner en formato horario las variables de radiación, ya que son las

únicas que siguen el modelo clear-sky.

Para comprobar la efectividad de dicho algoritmo se escogió una columna del conjunto de datos del modelo determinista correspondiente a una longitud y latitud de la península y a las cinco variables de radiación que se tenían descargadas. A continuación, se realizó el mismo procedimiento que en el experimento con el modelo determinista trihorario, con la diferencia de que ahora en vez de interpolar la producción obtenida, se interpolaron las variables de radiación.

En la Figura 5.2.1 se muestra en qué grado se ajusta la predicción de los valores horarios de cada variable tras aplicar la interpolación mediante clear-sky a los valores reales de cada una. Como se puede observar, las variables CDIR y SSRC obtienen una interpolación casi perfecta. Esto es así porque ambas son obtenidas a partir del clear-sky. También se observa que SSR y SSRD se parecen mucho, por tanto, solo una de las dos se usó como variable de los modelos, SSR.

Tras esta pequeña prueba se puede decir que sería viable interpolar las variables de radiación antes de entrenar los modelos y se propone como parte de los trabajos que podrían continuar este. Por otro lado, esta interpolación es probable que no se adapte a las variables que no son de radiación, por lo que habría que investigar otro tipo, como podría ser la interpolación cúbica.

5.3. Conclusiones finales

En este proyecto se ha realizado una fase de análisis de los distintos algoritmos que se han encontrado en la literatura para resolver problemas de predicción de la producción de energía fotovoltaica, como pueden ser SVR, Descenso Dual por Coordenadas o Pegasos. Finalmente, se escogió SVR por ser el más utilizado y el que a priori obtenía mejores resultados en este tipo de problemas para realizar los experimentos. Como lo que se quería estudiar era cómo afectaba el uso del modelo de ensembles del Centro Europeo a la predicción horaria, se implementó un algoritmo de interpolación basado en clear-sky para pasar los resultados en formato trihorario a formato horario.

Como conclusión final tras analizar los resultados obtenidos por todos los experimentos, se puede afirmar que el uso de los 50 ensembles meteorológicos no proporciona mejoría en los resultados de los modelos. Por tanto, se recomienda el uso del ensemble de control, que sí mejora de forma notable los resultados obtenidos con el modelo determinista trihorario. Un punto de interés futuro es usar los ensembles NWP para dar estimaciones de la incertidumbre asociada a las predicciones de energía.

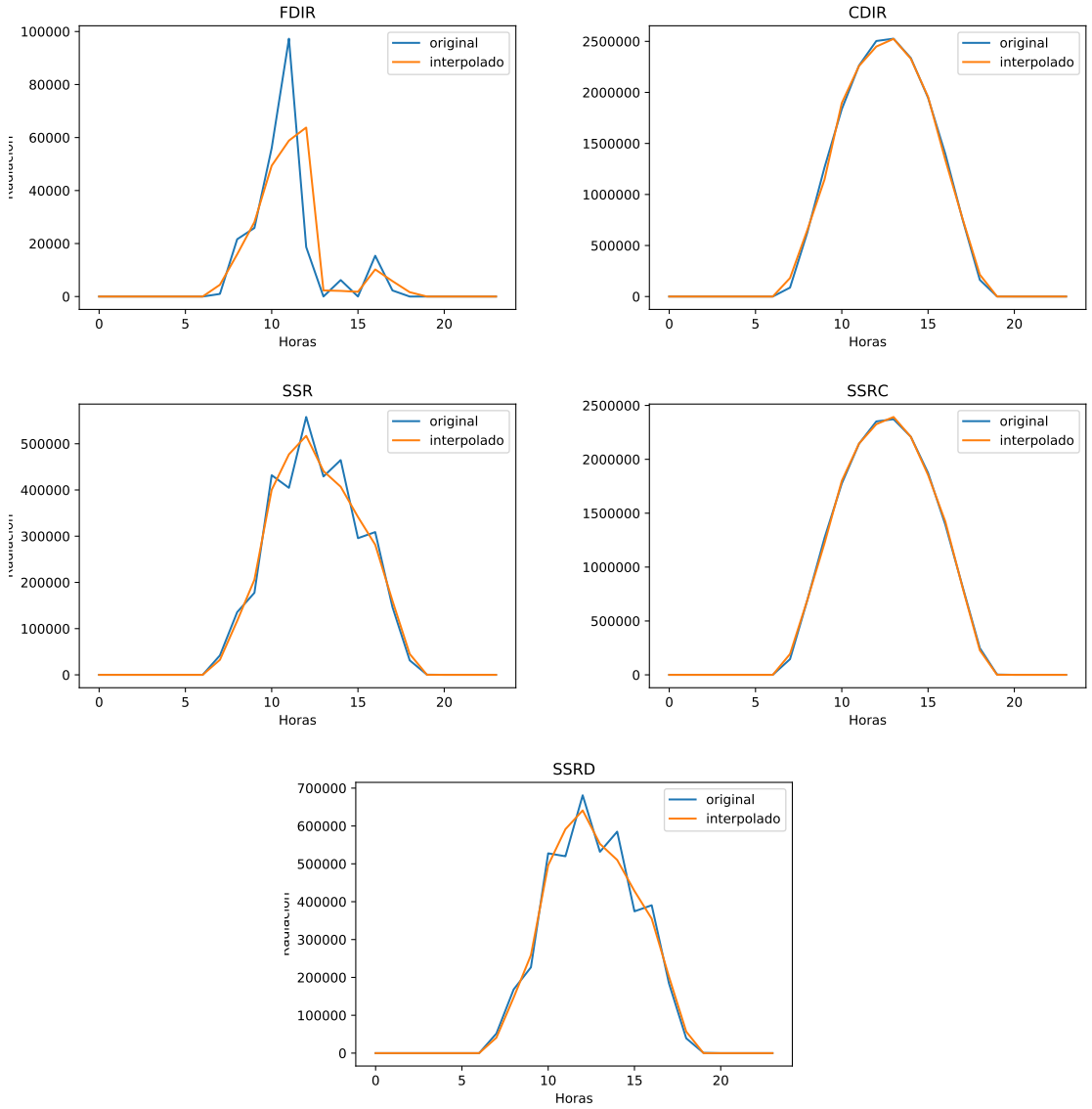


Figura 5.2.1: Comparación de los valores obtenidos mediante la interpolación con los valores horarios reales de cada una de las variables de radiación.

Bibliografía

[1] Diferencias entre energía termosolar y fotovoltaica. http://www.laenergiadelcambio.com/diferencias-entre-energia-termosolar-y-fotovoltaica-2. Accedido el 10-01-2018.

[2] Y.Gala, A.Fernández, J.Díaz, and J.R.Dorronsoro. Support vector forecasting of solar radiation values. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8073 LNAI:51–60, 2013.

[3] R.W.Andrews, J.S.Stein, C.Hansen, and D.Riley. Introduction to the open source PV LIB for python Photovoltaic system modelling package. In *2014 IEEE 40th Photovoltaic Specialist Conference, PVSC 2014*, pages 170–174. IEEE, jun 2014.

[4] J.Antonanzas, N.Osorio, R.Escobar, R.Urraca, F.J.Martinez de Pison, and F.Antonanzas-Torres. Review of photovoltaic power forecasting. *Solar Energy*, 136:78–111, 2016.

[5] C.Wan, J.Zhao, Y.Song, Z.Xu, J.Lin, and Z.Hu. Photovoltaic and solar power forecasting for smart grid energy management. *CSEE Journal of Power and Energy Systems*, 1(4):38–46, dec 2015.

[6] Who we are — ECMWF. https://www.ecmwf.int/en/about/who-we-are http://www.ecmwf.int/en/about/who-we-are. Accedido el 12-01-2018.

[7] Numerical weather prediction - Simple English Wikipedia, the free encyclopedia. https://simple.wikipedia.org/wiki/Numerical_weather_prediction. Accedido el 12-01-2018.

[8] Modelling and Prediction — ECMWF. https://www.ecmwf.int/en/research/modelling-and-prediction. Accedido el 12-01-2018.

[9] R.Buizza. The ECMWF ensemble prediction system. *Predictability of Weather and Climate*, 9780521848824:459–488, 2006.

[10] A.Mellit, A.Massi Pavan, and V.Lughi. Short-term forecasting of power production in a large-scale photovoltaic plant. *Solar Energy*, 105:401–413, jul 2014.

[11] N.Sharma, P.Sharma, D.Irwin, and P.Shenoy. Predicting Solar Generation from Weather Forecasts Using Machine Learning. *Smart Grid Communications (SmartGridComm), 2011 IEEE International Conference*, pages 528–533, 2011.

[12] J.Friedman T.Hastie, R.Tibshirani. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. 1994.

[13] J.Shawe-Taylor N.Cristianini. *Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. 2000.

[14] A.J.Smola and B.Sc Olkopf. A tutorial on support vector regression *. *Statistics and Computing*, 14:199–222, 2004.

[15] R.O.Duda, P.E.Hart, and D.G.Stork. *Pattern classification*. 2nd edition, 2001.

[16] A.Torres-Barrán and J.R.Dorronsoro. Conjugate descent for the SMO algorithm. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2016-Octob, pages 3817–3824, 2016.

[17] 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*, 23(7):1142–1147, 2012.

[18] T.Glasmachers. On related violating pairs for working set selection in SMO algorithms. *ESANN*, (April):475–480, 2008.

[19] C.Ho and C.Lin. Large-scale Linear Support Vector Regression. *Jmlr*, 13:3323–3348, 2012.

[20] C.Hsieh and C.Lin. A Dual Coordinate Descent Method for Large-scale Linear SVM. (2), 2008.

[21] R.Fan, K.Chang, C.Hsieh, X.Wang, and C.Lin. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9(2008):1871–1874, 2008.

[22] S.Shalev-Shwartz, Y.Singer, and N.Srebro. Pegasos: Primal Estimated sub-{GrAdient SOlver for SVM}. *Proc. {ICML}*, 2007.

[23] A.Catalina and J.R.Dorronsoro. NWP ensembles for wind energy uncertainty estimates. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10691 LNAI, 2017.

[24] J.Shi, W.Lee, Y.Liu, Y.Yang, and P.Wang. Forecasting power output of photovoltaic systems based on weather classification and support vector machines. *IEEE Transactions on Industry Applications*, 48(3):1064–1069, 2012.

[25] A.Fernández, Y.Gala, and J.R.Dorronsoro. Machine Learning Prediction of Large Area Photovoltaic Energy Production. pages 38–53. Springer, Cham, 2014.