

Grado en Estadística y Empresa  
2021-2022

*Trabajo Fin de Grado*

# “Predicción probabilística de la producción eólica en Cádiz”

---

Javier Cárdenas Serres

Tutor

Ricardo Aler Mur

## DETECCIÓN DEL PLAGIO

La Universidad utiliza el programa **Turnitin Feedback Studio** para comparar la originalidad del trabajo entregado por cada estudiante con millones de recursos electrónicos y detecta aquellas partes del texto copiadas y pegadas. Copiar o plagiar en un TFG es considerado una **Falta Grave**, y puede conllevar la expulsión definitiva de la Universidad.



Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No Comercial - Sin Obra Derivada**

## **RESUMEN**

Falta escribir resumen

## ÍNDICE GENERAL

1. INTRODUCCIÓN. . . . .	1
2. LA PRODUCCIÓN Y PREDICCIÓN DE ENERGÍA EÓLICA . . . . .	2
2.1. Energía eólica en España . . . . .	2
2.1.1. Ventajas e inconvenientes . . . . .	3
2.2. Modelo numérico para la predicción meteorológica . . . . .	3
3. PREDICCIÓN PROBABILÍSTICA . . . . .	4
3.1. Función de pérdida Smooth Pinball . . . . .	5
3.2. De las predicciones a la distribución . . . . .	6
3.3. Intervalos a predecir . . . . .	7
3.4. Métricas de evaluación. . . . .	7
3.4.1. Quantile Score . . . . .	7
3.4.2. Average Coverage Error . . . . .	7
3.4.3. Interval Score. . . . .	8
4. METODOLOGÍA Y FUENTE DE DATOS . . . . .	9
4.1. Fuente de datos. . . . .	9
4.1.1. Variable a predecir . . . . .	9
4.1.2. Variables de entrada . . . . .	9
4.1.3. Preproceso . . . . .	12
4.2. Metodología . . . . .	12
4.2.1. Regresión cuantílica . . . . .	12
4.2.2. Perceptrón multicapa . . . . .	13
4.2.3. Long short-term memory . . . . .	13
5. RESULTADOS . . . . .	16
5.1. Convergencia en el entrenamiento . . . . .	16
5.2. Resultados en el set de validación . . . . .	17
5.3. Optimización de hiperparámetros. . . . .	18
5.3.1. Número de capas ocultas . . . . .	19
5.3.2. Número de observaciones hacia atrás . . . . .	19

5.4. Otras pruebas de interés . . . . .	20
5.4.1. Smooth Pinball vs Pinball . . . . .	20
5.4.2. Spread vs Non Spread . . . . .	20
5.4.3. Todos los datos vs la propia variable predictiva . . . . .	21
6. CONCLUSIONES . . . . .	22

# 1. INTRODUCCIÓN

Desde que se implantó por primera vez un aerogenerador en 1984, la energía eólica ha ido aumentando su capacidad productiva, hasta ser en 2021 la principal fuente de electricidad en España.

Sin embargo, a pesar de ser una fuente de energía limpia y sostenible, las corrientes de aire son variantes y poco estables, haciendo que la producción de energía eólica sea difícil de predecir e impidiendo establecer una oferta óptima en la subastas de electricidad que se llevan a cabo diariamente en España.

El principal objetivo de este trabajo es desarrollar un algoritmo que permita hacer predicciones probabilísticas sobre la producción de energía eólica que, al contrario que la predicción puntual, permita obtener la distribución de la producción futura esperada, aportando más información sobre el valor esperado y mejorando la toma de decisiones respecto a la cantidad a ofertar por las productoras de energía eólica.

Si bien hay diferentes maneras de hacer predicciones probabilísticas, en este trabajo en concreto, nos centraremos en la predicción cuantílica, de la que podremos sacar intervalos y estimar los posibles rangos en los que se debería encontrar la variable estimada en el futuro. Estas predicciones de generación de electricidad se harán para toda la región de Cádiz.

El trabajo está estructurado de la siguiente manera: En 2 se analiza la evolución y el papel de la energía eólica en España, además del sistema de predicción de las variables meteorológicas y el organismo encargado. En 3 se hace una descripción de la predicción probabilística, así como las funciones de pérdida y las métricas utilizadas. En la siguiente sección 4 se detalla la fuente de datos y la metodología utilizada para este trabajo. En 5 se comparan los resultados obtenidos por los diferentes algoritmos utilizados y se exponen los experimentos realizados. Por último, en 6 se resumen las principales conclusiones del estudio y el trabajo futuro.

## 2. LA PRODUCCIÓN Y PREDICCIÓN DE ENERGÍA EÓLICA

En este apartado veremos algunas de las características más importantes de la generación de energía eólica y algunos de los métodos utilizados para sus predicciones.

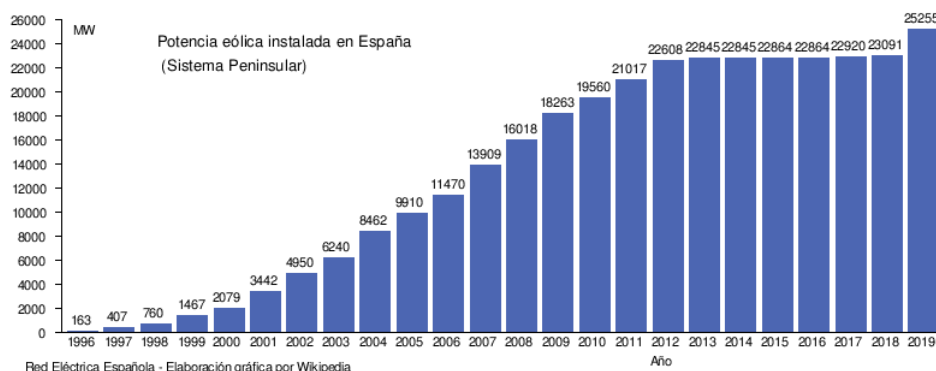
### 2.1. Energía eólica en España

La energía eólica es aquella que se obtiene a partir de las corrientes de aire. Actualmente esto se consigue a través de los aerogeneradores, transformando la energía cinética en energía mecánica. Es, además, una energía limpia, sostenible y se encuentra dentro de la categoría de energías renovables.

Desde que la cooperativa Ecotècnia instalase el primer aerogenerador en España en 1984 como posible alternativa a la energía nuclear hasta 2012, la eólica ha tenido un crecimiento constante, como se puede ver en la Figura 2.1. ....

**Figura 2.1**

*Evolución de la potencia eólica instalada*



Fuente: Wikipedia

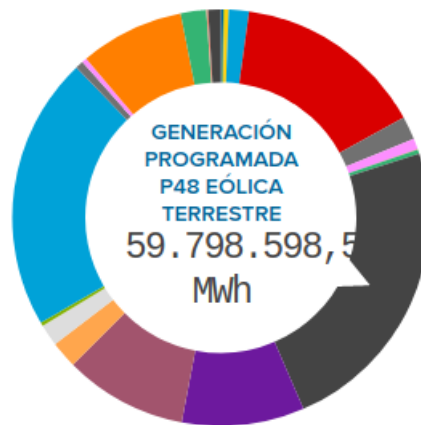
En 2021 la eólica pasó a ser la principal fuente de producción eléctrica en España con un porcentaje del 23 % sobre el total producido y representando mitad de la energía producida por las renovables, que fueron un 46 % en 2021, ver Figura 2.2. ....

citar historia de aerogenerador.

Citar fuente red eléctrica.

**Figura 2.2**

*Generación de electricidad eólica respecto al resto de fuentes en 2021*



### 2.1.1. Ventajas e inconvenientes

Uno de los mayores problemas es que, a pesar de ser actualmente la principal fuente de energía en España, su producción es bastante impredecible ya que depende de las características temporales del momento y estas a su vez son muy difíciles de estimar. Esto afecta directamente en la toma de decisiones a la hora de saber cuánta cantidad de energía ofertar para el día siguiente en las subastas de electricidad que se hacen diariamente en España. ...

Añadir cita.

### 2.2. Modelo numérico para la predicción meteorológica

El matemático Lewis Fry Richardson fue la primera persona en proponer un modelo numérico para la predicción meteorológica (NWP, por sus siglas en inglés) en 1922 .... No obstante, no fue hasta el inicio de los ordenadores en 1950 cuando se pudo hacer por primera vez las primeras predicciones meteorológicas utilizando métodos numéricos simples. ....

citarle con su libro Weather prediction numerical process.

Desde entonces, el avance de los ordenadores ha permitido que se mejoren las técnicas de predicción de variables meteorológicas y con ellas la precisión de dichas predicciones.

Como explicaremos en más detalle en 4, utilizaremos como fuente de información las predicciones meteorológicas del Centro Europeo de Previsiones Meteorológicas a Plazo Medio (ECMWF, por sus siglas en inglés). ....

citar fuente.

Añadir referencia.

### 3. PREDICCIÓN PROBABILÍSTICA

En esta sección se definirá la predicción probabilística y se detallarán algunos de las ventajas más importantes respecto a las predicciones puntuales.

Al contrario que la predicción puntual, la predicción probabilista aporta más información permitiendo obtener la distribución de la producción futura y mejorando la toma de decisiones respecto a la cantidad a ofertar por las productoras de energía. . . .

Si bien existen diferentes maneras de hacer predicciones probabilísticas, en este trabajo nos centraremos en la predicción cuantílica para luego poder sacar la distribución de la variable a predecir.

Fue en 1978 cuando, en busca de una alternativa robusta al estimador de mínimos cuadrados, Koenker y Basset desarrollaron una nueva regresión lineal que tuviese en cuenta, en vez de la media, la mediana por ser un estimador robusto y no dependiente de los valores extremos que pueda existir en una muestra. . . .

A esta nueva regresión la llamaron la regresión cuantílica. La regresión cuantílica tiene como función dado el error 3.1, minimizar la función 3.2.

$$\xi = y - \hat{y} \quad (3.1)$$

$$p_{\tau}(\xi) = \begin{cases} \tau\xi & \text{si } \xi \geq 0, \\ (\tau - 1)\xi & \text{si } \xi < 0. \end{cases} \quad (3.2)$$

Donde  $\tau$  es el cuantil que se quiere predecir. La ecuación 3.2, también se la suele llamar función de coste Pinball.

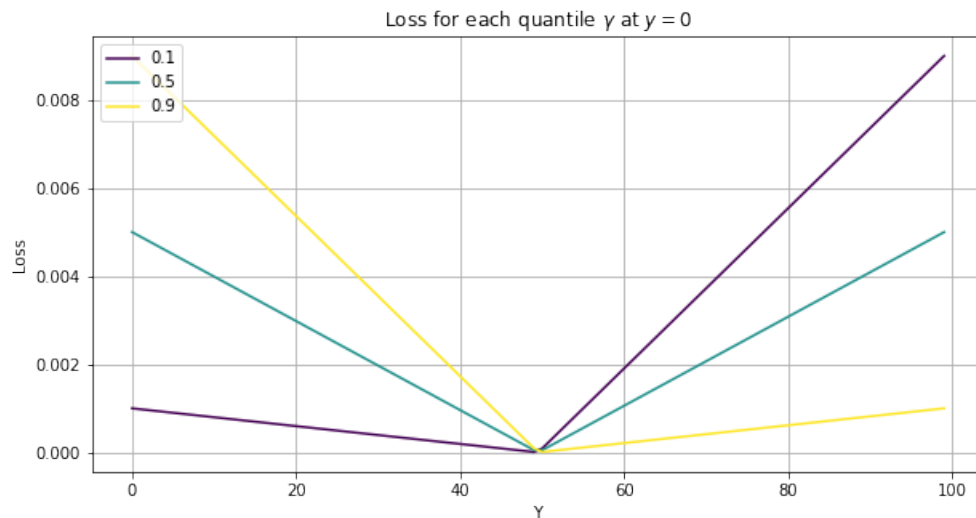
Como podemos ver en la Figura 3.1, la función penaliza más cuando el cuantil alto/bajo predicho se encuentra por debajo/encima de la variable observada para  $y = 0$  y los cuantiles 0,1 y 0,9.

Añadir  
referencia  
de los  
dos  
estudios.

Añadir  
referencia  
de  
Koenker.

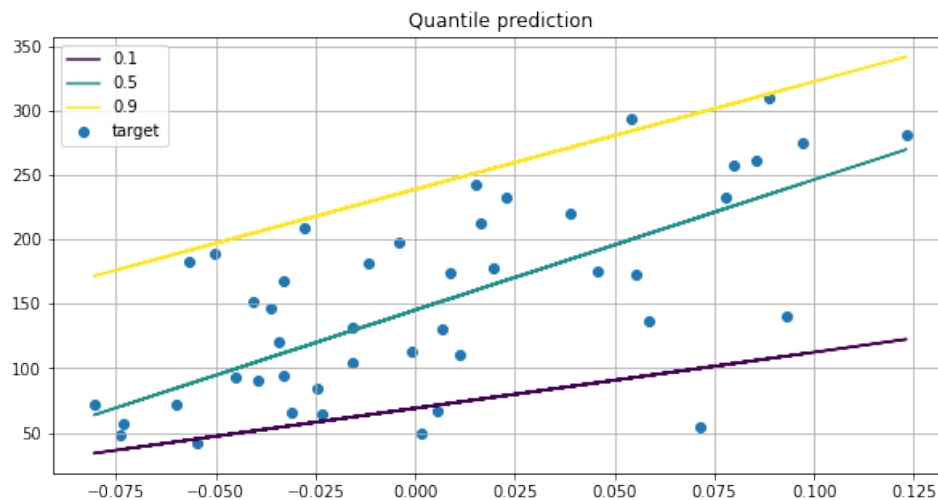


**Figura 3.1**  
*Función de pérdida Pinball*



Y en la Figura 3.2 se muestra un ejemplo de cómo quedaría una predicción de la regresión cuantílica para los cuantiles 0,1, 0,5 y 0,9.

**Figura 3.2**  
*Ejemplo de la predicción de cuantiles*



### 3.1. Función de pérdida Smooth Pinball

Uno de los problemas de la función de pérdida utilizada para la regresión cuantílica es que no es derivable en 0 y por tanto no se puede utilizar en modelos que utilizan el descenso de gradiente para optimizar los pesos... Por esta razón en . . . se desarrolla una nueva función de coste que permite ser utilizada en modelos que utilicen el descenso de gradiente.

Añadir  
referencia.

Añadir  
referencia  
del  
estu

Una de las ventajas de esta función es que nos permite ponerla en modelos de redes neuronales además de los típicos modelos simples como la regresión lineal ...

Añadir  
cita.

La función de coste Smooth Pinball quedaría definida como:

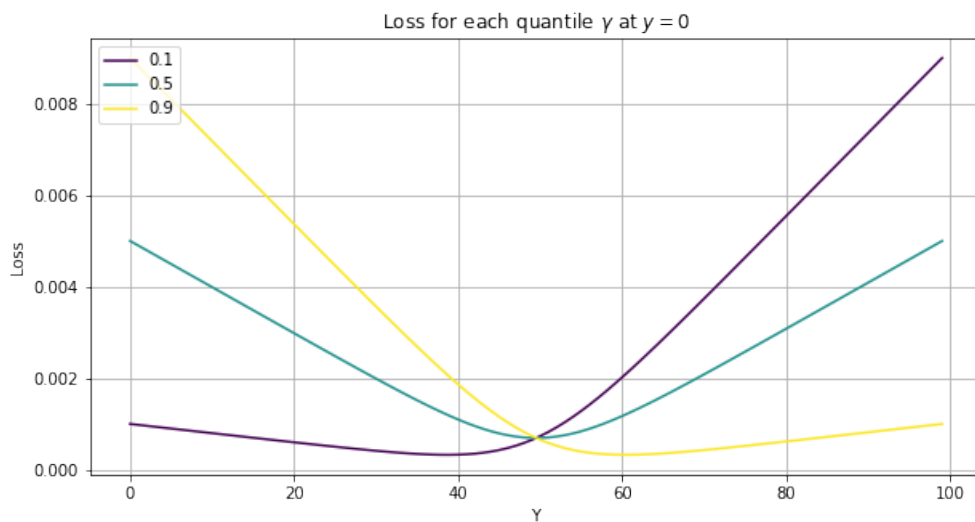
$$S_{\tau,\alpha}(\xi) = \tau\xi + \alpha \log(1 + e^{-\frac{\xi}{\alpha}}) \quad (3.3)$$

Donde  $\alpha > 0$  es el parámetro smoothing.

En la Figura 3.3 se muestra un ejemplo del coste obtenido por un cuantil y una predicción dada.

**Figura 3.3**

*Función de pérdida Smooth Pinball*

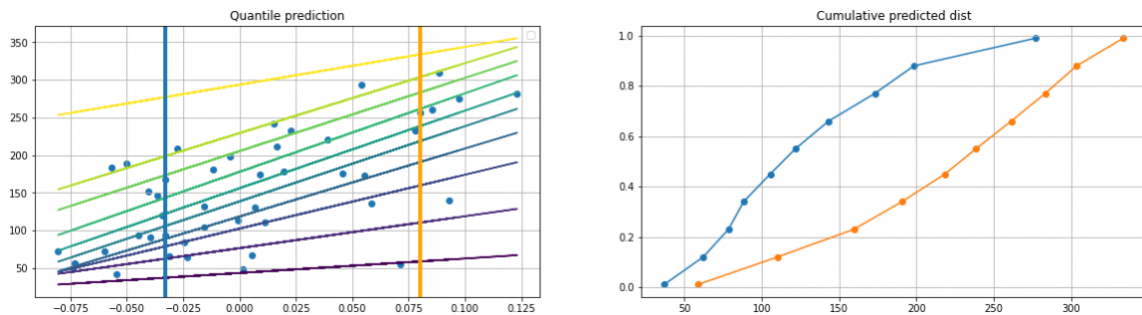


### 3.2. De las predicciones a la distribución

Juntando las predicciones cuantílicas podemos generar una función de distribución acumulada de la variable a predecir como aparece en la siguiente figura.

**Figura 3.4**

*Función de distribución acumulada a partir de las predicciones*



### 3.3. Intervalos a predecir

Si bien se han probado a predecir diferentes rangos, finalmente en este trabajo se van a predecir los siguientes intervalos: 95 %, 90 %, 80 % y 70 %

Para hacer la predicción de dichos intervalos, se van a predecir los siguiente cuantiles: 0,025; 0,05; 0,1; 0,15; 0,85; 0,9; 0,95 y 0,975.

### 3.4. Métricas de evaluación

Para las predicciones de los modelos será importante tener una serie de métricas que permitan evaluar correctamente cada algoritmo. Para ello utilizaremos métricas que no solo evalúen las estimaciones de los cuantiles sino también los intervalos predichos a partir de esos cuantiles. Las métricas utilizadas son las mismas que en el estudio .... Estas son:

Citar  
estu-  
dio.

#### 3.4.1. Quantile Score

Esta primera métrica permite evaluar la bondad de las estimaciones de los cuantiles. Utilizando la función de pérdida Pinball para todas las muestras obtenemos el quantile score (QS) y será la métrica principal que utilizaremos para evaluar los modelos. Cuanto menor sea esta métrica, mejores serán las predicciones.

$$QS = \sum_{t=1}^N \sum_{m=1}^M p_{tm}(y_t - \hat{q}_t^{\tau_m}) \quad (3.4)$$

#### 3.4.2. Average Coverage Error

Por otro lado, también necesitaremos evaluar la bondad de los intervalos creados a partir de las predicciones de los cuantiles. Para ello se utilizarán tres métricas ACE, IS y Sharpness.

La métrica ACE, mide cuan cerca certeros son los intervalos predichos según las variables observadas. Como ejemplo, si utilizásemos un intervalo del 80 %, idealmente queríamos que los datos observados estuvieran fuera de dicho intervalo un 20 % de las veces.

A continuación se muestra cual es la funciones que determina dicha métrica. Cuanto más cerca de 0 este el valor de la métrica ACE, mejores serán los resultados del modelo.

$$PICP_i = \frac{1}{N} \sum_{t=1}^N \mathbb{1}_{y_t \in I_t^{\beta_i}(x_t)} \quad (3.5)$$

$$ACE = \sum_{i=1}^{\frac{M}{2}} |PICP_i - 100(1 - \beta_i)| \quad (3.6)$$

### 3.4.3. Interval Score

El problema de la métrica anterior es que en ocasiones se podrían obtener buenos resultados simplemente haciendo los intervalos más anchos. La métrica Interval Score (IS) y *Sharpness* permiten evaluar al modelo según la anchura de sus intervalos. A continuación en 3.7 y 3.8 se muestran las funciones necesarias para calcular dichas métricas.

Si bien, a la hora de evaluar los resultados nos centraremos en el IS más que en el *sharpness*, se incluirá también esta última en las tablas a modo informativo.

$$Sharpness = \frac{2}{NM} \sum_{t=1}^N \sum_{i=1}^{\frac{M}{2}} (u_t^{\beta_i} - l_t^{\beta_i}) \quad (3.7)$$

$$IS = Sharpness + \frac{2}{\beta_i} (l_t^{\beta_i} - y_t) \mathbb{1}(y_t < l_t^{\beta_i}) + \frac{2}{\beta_i} (y_t - u_t^{\beta_i}) \mathbb{1}(y_t > u_t^{\beta_i}) \quad (3.8)$$

## 4. METODOLOGÍA Y FUENTE DE DATOS

### 4.1. Fuente de datos

Los datos utilizados en el trabajo se han sacado de dos fuentes diferentes:

1. *ESIOS*: Es el sistema de información desarrollado por Red Eléctrica que contiene la información relevante sobre el mercado eléctrico en España.
2. *ECMWF*: El Centro Europeo de Previsiones Meteorológicas (ECMWF, por sus siglas en inglés).

#### 4.1.1. Variable a predecir

La variable a predecir es la generación de energía eólica horaria en la región de Cádiz desde 2015 a 2018. La razón de utilizar este intervalo es que, durante este periodo, la capacidad instalada de eólica se mantuvo constante y por tanto no es necesario hacer ningún tipo de ajuste en los datos.

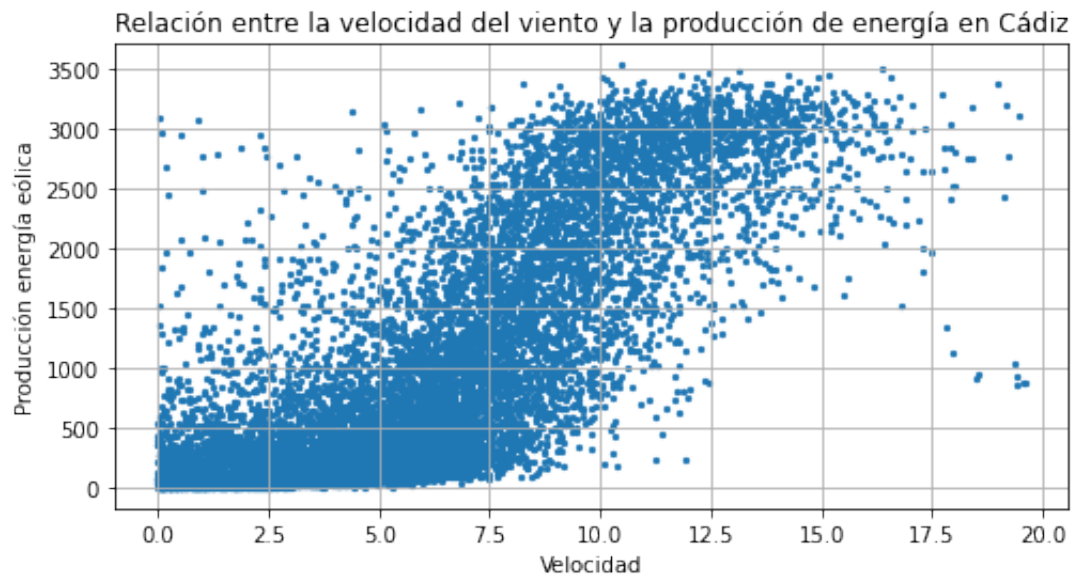
#### 4.1.2. Variables de entrada

Las variables de entrada se han sacado de ECMWF y son:

- *10 y 100m u-component of wind*: La velocidad, en metros por segundo, del movimiento del aire en dirección este a 10 y 100m sobre la superficie terrestre.
- *10 y 100m v-component of wind*: La velocidad, en metros por segundo, del movimiento del aire en dirección norte a 10 y 100m sobre la superficie terrestre.
- *2m temperature*: Es la temperatura del aire a dos metros sobre el nivel del mar.
- *Surface pressure*: La presión de la atmósfera en la superficie terrestre.
- *Norma de la velocidad a 10m y 100m*: Es una variable calculada a partir de las variables u-component y v-component of wind. Se calcula de la siguiente forma, donde  $i = 10, 100$ :
$$n_i = \sqrt{u_i^2 + v_i^2}$$
- *Hora, día y mes*: Se han añadido 3 variables temporales por si pudiesen resultar de interés.

**Figura 4.1**

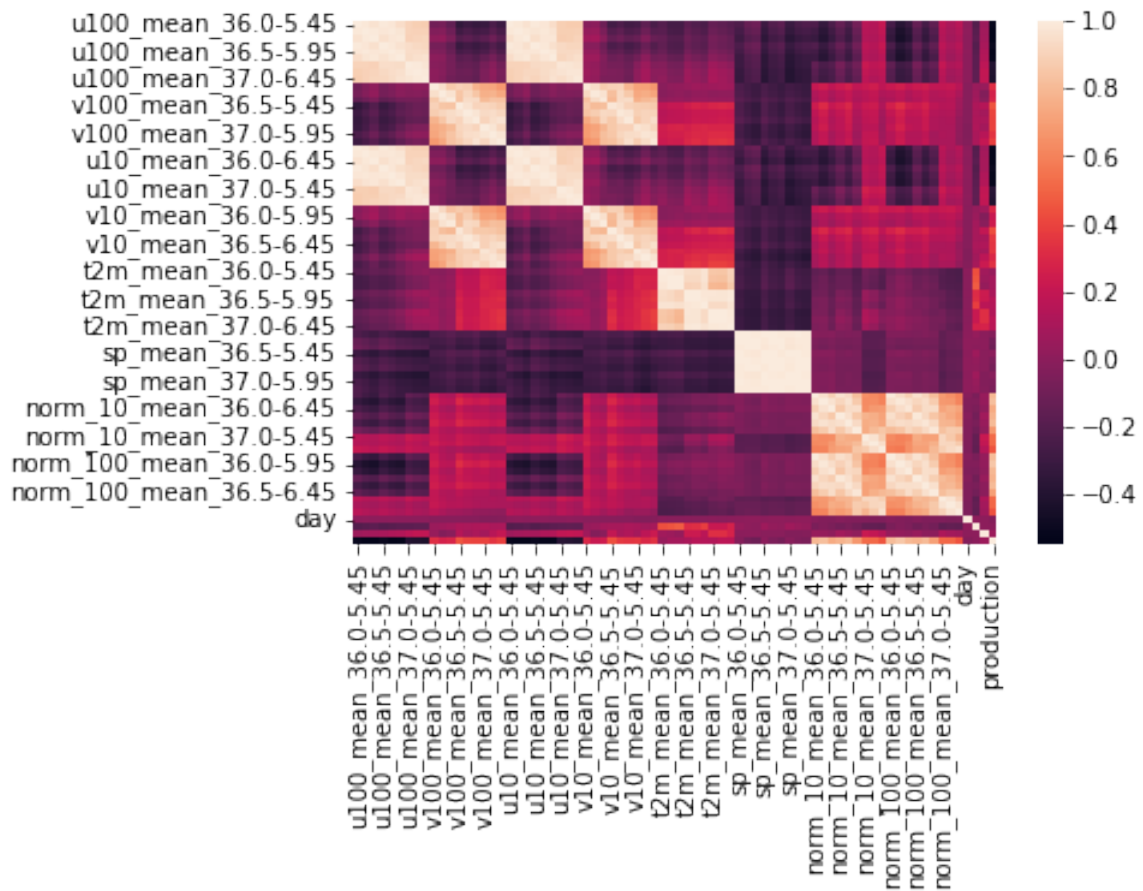
*Relación entre la velocidad del viento y la energía producida*



ECMWF nos da los valores (reales y estimados) de estas variables por cuadrantes de 30km. Teniendo esto en cuenta, para la región de Cádiz hay 9 cuadrantes por lo que tendremos 9 estimaciones de las variables mencionadas para cada intervalo de tiempo. Esto hace un total de 75 variables de entrada  $((9 * 8) + 3 = 75)$ .

**Figura 4.2**

*Gráfica de correlaciones entre las variables de entrada*



## Mean y Spread

Recordemos que las variables que se encuentran en el ECMWF son el resultado de una combinación de predicciones. Actualmente ECMWF permite obtener tanto la media como la desviación de la combinación de sus modelos.

Las 75 variables mencionadas anteriormente solo son asumiendo que estamos utilizando la media de los modelos. No obstante, podría pensarse en añadirse también las variables de desviación, lo que añadiría 36 variables nuevas ( $9 * 6 = 36$ , quitando las dos normas) al total de variables de entrada.

En principio la combinación de ambos estimadores (tanto la media como la desviación de la combinación de los modelos) parece ser interesante, así que se ha decidido en este trabajo también dedicar un apartado (ver Sección 5.4.2) a la comprobación y al análisis de la bondad de estos estimadores.

### 4.1.3. Preproceso

Como hemos mencionado anteriormente, los datos de producción eólica sacados de ESIOS son cada hora, mientras que las variables que se utilizan para predecir el viento son cada tres. Por este motivo, se han agrupado los datos de la producción eólica cada tres horas, sumando durante ese intervalo toda la energía generada.

Una vez hemos sincronizado las variables de entrada con la variable a predecir, haremos una separación entre los datos de entrenamiento, los datos de validación y los datos de test (80/10/10 respectivamente).

Cabe mencionar que se han normalizado los datos de entrada para acelerar y mejorar el entrenamiento de la red. ...

Citar:  
Nor-  
mali-  
zation  
Tech-  
niques  
in  
Trai-  
ning  
DNNs:  
Metho-  
do-  
logy,  
Analy-  
sis and  
Appli-  
cation.

## 4.2. Metodología

En este apartado detallaremos los algoritmos utilizados para hacer las predicciones.

De menor a mayor complejidad tendremos los siguientes modelos:

- **Regresión cuantílica:** este será nuestro algoritmo de referencia y veremos si podemos mejorarlo. Se trata de un modelo simple, lineal. A partir de este modelo podremos ver si el aumento en la complejidad de los algoritmos es justificable.
- **Perceptrón multicapa:** Como primera acercamiento a las redes neuronales y a algoritmos no lineales, se probará una red con múltiples capas.
- **LSTM:** Por último, se utilizará una red recurrente, en concreto la LSTM por su idoneidad con el problema a resolver.

Por simplicidad se ha probado primero una única versión de cada algoritmo (sin hacer ningún cambio en los hiperparámetros). Después de comparar los resultados entre los tres tipos de modelo se ha procedido a hacer la optimización de los hiperparámetros del mejor de ellos.

### 4.2.1. Regresión cuantílica

Como hemos mencionado anteriormente, la regresión cuantílica fue desarrollada por Koenker y Bassett con el objetivo de evitar hacer asunciones sobre la distribución de los errores de una regresión lineal...

Este modelo será el más sencillo que utilicemos y lo utilizaremos como base para comparar los modelos más complejos y evaluar si finalmente el aumento de complejidad en el modelo está justificado y respaldado por una mejora en las métricas de evaluación con respecto al modelo de regresión lineal cuantílica.

Citar  
estu-  
dio.



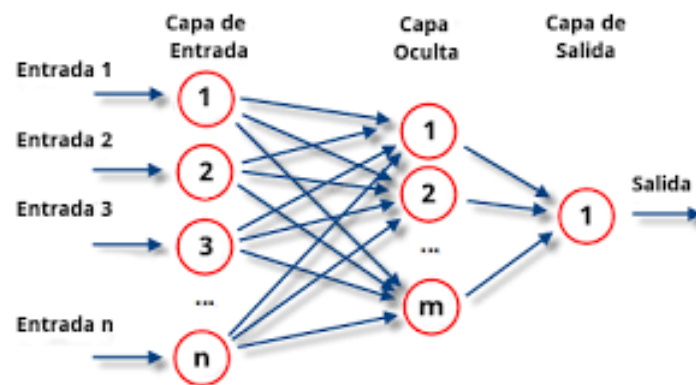
Al igual que los siguientes dos modelos, se ha desarrollado este algoritmo en *PyTorch* y se ha utilizado el descenso de gradiente para conseguir los pesos óptimos.

#### 4.2.2. Perceptrón multicapa

El perceptrón multicapa es un tipo de red neuronal formado al menos por tres capas: La capa de entrada, la capa oculta y la capa de salida, como se puede venir en la siguiente Figura:

**Figura 4.3**

*Ejemplo de estructura de un perceptrón multicapa*



Fuente: Wikipedia

A diferencia del modelo anterior, el perceptrón multicapa es un algoritmo más complejo y puede utilizarse para datos que no son linealmente separables.

Este algoritmo tiene los siguientes hiperparámetros:

- Número de capas ocultas: Para este trabajo se ha fijado a 2.
- Número de neuronas por capa: El número de neuronas por cada capa serán 50 y 20 respectivamente. La razón de que se haya fijado a 50 el número de neuronas en la primera capa oculta, teniendo en cuenta que las variables de entrada son 75 es para ver si la red es capaz de reducir las dimensiones sin perder la información y al mismo tiempo evitando tener que hacer un PCA o un algoritmo similar para reducir la dimensionalidad de las variables de entrada.
- Optimizador a utilizar: Se ha utilizado el algoritmo de *Adam*, con un learning rate de 0.001. Y las betas que da por defecto la función de optimización de *PyTorch*.

#### 4.2.3. Long short-term memory

Por último utilizaremos un algoritmo más complejo, la LSTM. Las LSTM forman parte del conjunto de redes neuronales recurrentes. Fueron introducidas por primera vez

por Hochreiter y Schmidhuber ... en 1997 y fueron adaptadas y mejoradas posteriormente por un conjunto amplio de personas. ...

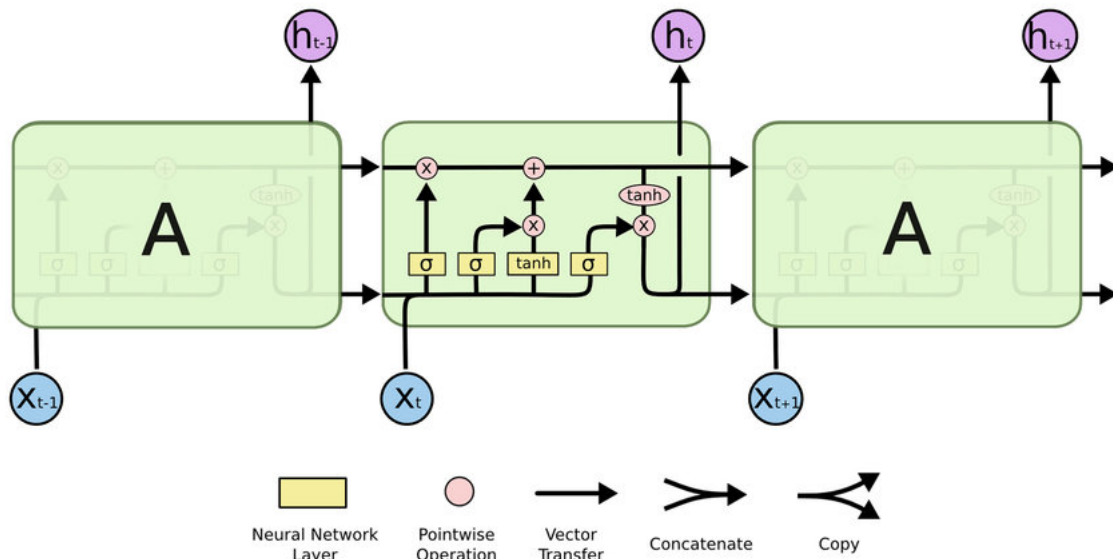
Citar estudio LSTM.

Se ha decidido introducir este algoritmo en el trabajo dada su capacidad de aprender relaciones a largo plazo, especialmente en problemas de predicción secuencial...

Citar estudio LSTM.

**Figura 4.4**

*Ejemplo de estructura de una LSTM*



Citar estudio LSTM en series temporales.

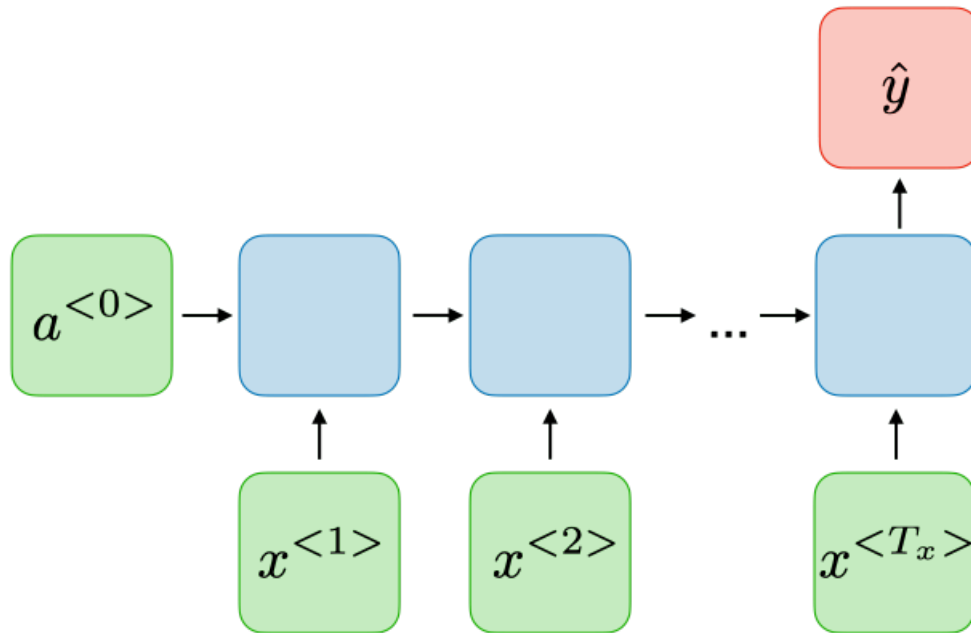
Fuente: Colah's Blog

Debido a la naturaleza de este algoritmo y sabiendo que es necesario pasarle una secuencia, se ha decidido utilizar un enfoque many-to-one ..., esto es, introducir en el input del modelo una secuencia de los datos con los  $n$  datos anteriores y tratar de predecir un único valor, en este caso un cuantil.

Citar estudio seq2point.

**Figura 4.5**

*Ejemplo de enfoque many-to-one*



Fuente: Stanford.edu

Como en los algoritmos anteriores, en la LSTM, por simplicidad, solo se probará una única combinación de hiperparámetros y en el caso de que sea el algoritmo que mejores resultados obtenga, se hará una optimización de hiperparámetros.

- Número de capas ocultas: Para este trabajo se ha fijado a 2.
- Número de estados ocultos: Se ha fijado al mismo número que las variables de entrada.
- Optimizador a utilizar: Se ha utilizado el algoritmo de *Adam*, con un learning rate de 0.0001 y las mismas betas que se ponen por defecto en *PyTorch*.
- Número de observaciones hacia atrás: Otra de las variables importantes a tener en cuenta en la LSTM es el número de datos que le pasamos hacia atrás. En la primera prueba se ha decidido fijar a 6, esto es, utilizará los últimos 6 datos observados de cada uno de los inputs para predecir la producción. Como nuestros datos son cada 3 horas, miraremos los valores de las variables meteorológicas 18h hacia atrás.

## 5. RESULTADOS

En este apartado se analizarán los resultados obtenidos por los modelos probados.

### 5.1. Convergencia en el entrenamiento

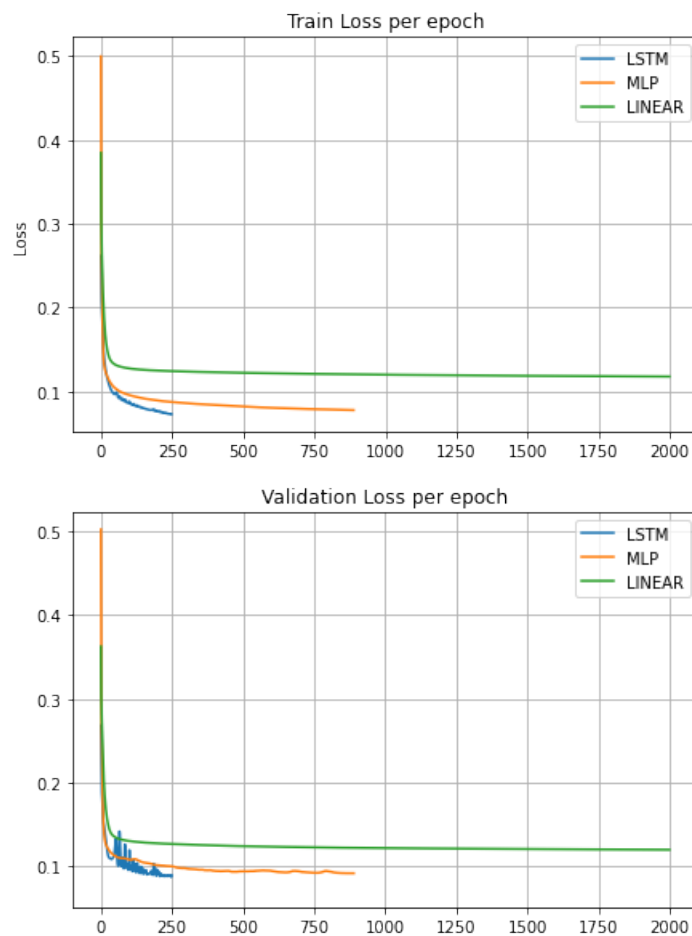
Los tres algoritmos parecen converger en el entrenamiento y no parecen estar sobreajustados.

A pesar de que hablaremos del set de validación en el siguiente apartado, se ha decidido añadir la gráfica de coste en el set de validación porque hemos utilizado early stopping ... esto es, el entrenamiento se para en el momento que el coste en el set de validación empieza a aumentar.

Citar estudio early stopping.

**Figura 5.1**

*Evolución de la pérdida en el entrenamiento*



En el entrenamiento los algoritmos MLP y LSTM parecen obtener mejores resultados que la regresión cuantílica, indicando que pueden existir relaciones no lineales entre las

variables de entrada y la variable de salida.

Otra data interesante es que a medida que se utiliza un modelo más complejo, suele parar antes el entrenamiento (los modelos complejos tienden más rápido a sobreajustar ...)

añadir  
refe-  
rencia

## 5.2. Resultados en el set de validación

A continuación se muestran los resultados obtenidos en el set de validación.

No se hacen comprobaciones en el test set ya que serán las últimas en hacerse cuando hayamos concluido con qué modelo quedarnos.

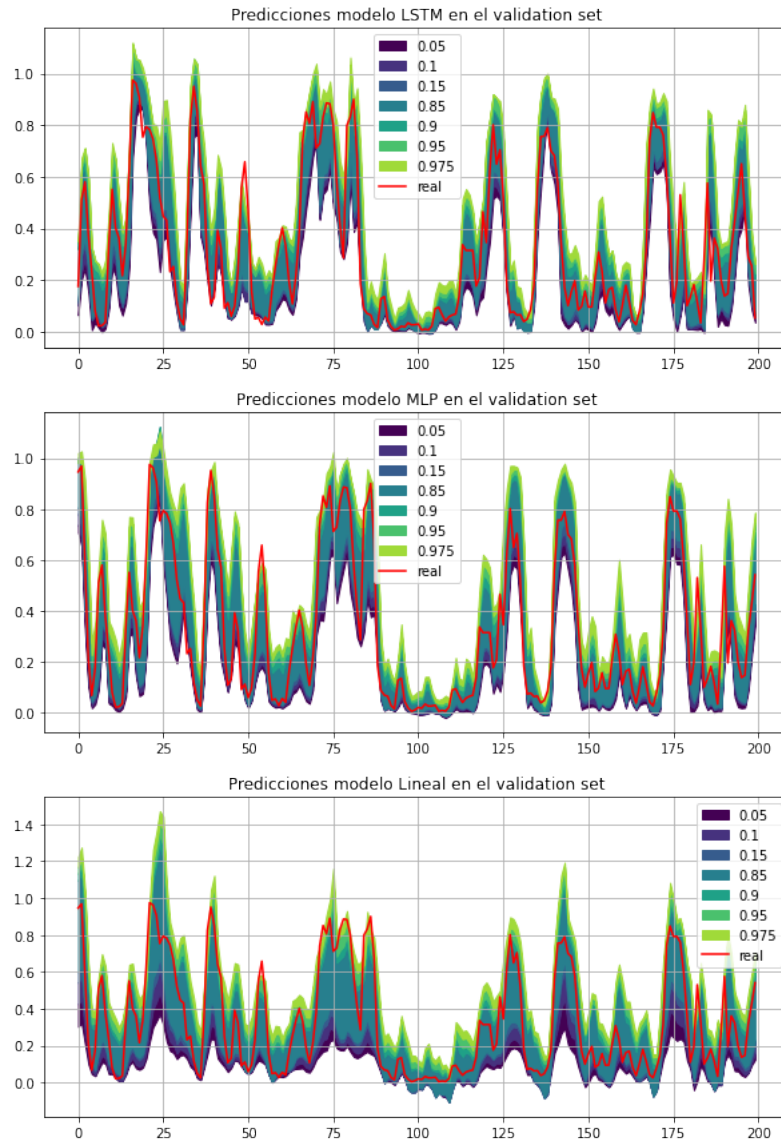
Como podemos ver, la LSTM parece obtener muy buenos resultados en el set de validación, junto con el modelo MLP. Esto no es de sorprender si tenemos en cuenta que se trata de un algoritmo específicamente diseñado para secuencias de datos y funciona muy bien en series temporales. ...

Si bien es cierto que tanto la LSTM como el modelo MLP tienen los dos unos resultados muy parecidos, hemos decidido continuar la investigación con la LSTM por sus mejores resultados en la métrica principal QS, además de tener mejores valores en las métricas IS y QS (además del *sharpness*).

Citar  
estu-  
dio  
bon-  
dades  
LSTM.

**Figura 5.2**

*Predicciones de los modelos en el set de validación*



Métrica	Regresión cuantílica	MLP	LSTM
QS	0.113	0.087	0.085
IS	0.381	0.303	0.300
Sharpness	0.299	0.201	0.181
ACE	1.567	2.051	4.537

### 5.3. Optimización de hiperparámetros

Como hemos visto en el apartado anterior, los resultados parecen indicar que el modelo que mejor funciona en estos casos es la LSTM.

En este apartado haremos una optimización de los hiperparámetros de la red y veremos que combinación de hiperparámetros es la que mejores resultados obtiene.

Debido al alto coste de entrenamiento, solo se van a tratar de optimizar dos parámetros:

- *El número de capas ocultas:* Se probará 1, 2 y 3 capas ocultas.
- *El número de observaciones hacia atrás:* 6, 16 y 24 observaciones. Esto observaría las 18h, 24h y 48h hacia atrás respectivamente.

### 5.3.1. Número de capas ocultas

Como se muestra en la siguiente tabla, el aumento de la complejidad, esto es, el aumento del número de capas ocultas, no parece ir acompañado de una mejora en los resultados.

Se puede ver, no obstante, como la LSTM con una capa obtiene unos resultados similares a los obtenidos con 3 capas. No obstante nos hemos decantado por seleccionar el modelo con una capa oculta por ser el más sencillo entre los dos.

Métrica	LSTM 1 capa	LSTM 2 capas	LSTM 3 capas
QS	0.085	0.132	0.087
IS	0.304	0.516	0.303
Sharpness	0.165	0.128	0.172
ACE	12.720	27.205	3.814

### 5.3.2. Número de observaciones hacia atrás

Una vez tenemos el número de capas que vamos a utilizar en la LSTM, también trataremos de optimizar el hiperparámetro *número de días hacia atrás*.

Como podemos ver, parece que añadir más información no aporta mejora en el modelo, siendo el modelo que mejores resultados tiene el que solo tiene 6 observaciones anteriores (la información de las 18h anteriores).

Métrica	LSTM 1 capa	LSTM 2 capas	LSTM 3 capas
QS	0.093	0.127	0.125
IS	0.341	0.502	0.448
Sharpness	0.156	0.166	0.209
ACE	8.621	21.949	28.376

Por tanto nuestro modelo final será una LSTM con una capa oculta y que mira únicamente 6 observaciones hacia atrás.

## 5.4. Otras pruebas de interés

Además de las pruebas mencionadas hasta ahora, se han realizado una serie de comprobaciones y análisis que se consideraba necesario para evaluar la bondad y la capacidad de los algoritmos propuestos.

### 5.4.1. Smooth Pinball vs Pinball

En un primer lugar, se ha tratado de comparar las diferencias en los resultados que puede ocasionar utilizar una función de pérdida u otra. Como se ha mencionado, la función de pérdida Pinball no es diferenciable en 0, por lo que podría dar problemas en la propagación *backward* a la hora de actualizar los pesos, no obstante es difícil que esto ocurra y por lo general no debería haber problemas en caso de utilizar una u otra.

A continuación se muestran los resultados obtenidos por el algoritmo LSTM (se ha decidido utilizar únicamente este algoritmo porque era el que mejores resultados obtenía).

Como podemos ver la LSTM que utiliza la función de pérdida Smooth Pinball parece obtener mejores resultados que su homóloga no diferenciable Pinball.

Métrica	Pinball	Smooth Pinball
QS	0.094	0.0898
IS	0.326	0.311
Sharpness	0.215	0.211
ACE	2.581	1.697

### 5.4.2. Spread vs Non Spread

Como hemos mencionado en el apartado 4.1.2, trataremos de analizar si existe alguna mejora en los resultados del modelo si utilizamos, además del estimador de la media de los modelos, el spread, esto es, su desviación.

En principio la desviación entre modelos podría ser interesante para el algoritmo ya que permite podría ampliar el intervalo cuantílico cuando existe una mayor discrepancia entre las predicciones de las variables de los modelos.

Debido al elevado número de variables (superando los 100 variables de entrada) se ha decidido realizar esta prueba con el algoritmo MLP que permite captar relaciones no lineales pero no es menos costoso computacionalmente que la LSTM.

Como se muestra en la Tabla 5.4.2, los resultados no parecen indicar que exista una mejora sustancial entre utilizar los datos de spread y no utilizarlos.



Métrica	Spread	Non Spread
QS	0.089	0.089
IS	0.306	0.309
Sharpness	0.205	0.214
ACE	1.194	1.404

#### 5.4.3. Todos los datos vs la propia variable predictiva

Una vez visto que la LSTM era el modelo que mejores resultados obtenía, también se ha probado a utilizar la producción de energía eólica como única variable para determinar si las variables de los modelos aportan información relevante.

Esta prueba, sin embargo, no podría llevarse a cabo en la realidad, ya que los datos de producción eólica se tienen con un día de retraso, por lo que las predicciones serían menos acertadas. No obstante, nos sirve como referencia de los resultados que podríamos esperar si tuviéramos los datos de producción con tres horas de retraso.

Como podemos ver, utilizando todos los datos de ECMWF se obtienen mejores resultados que solo utilizando un modelo propia serie de producción para predecirse a si misma. Esto nos permite afirmar que las variables de ECMWF sí que parecen estar aportando información al modelo sobre la generación futura de energía eólica.

Métrica	Autorregresivo	Todos los datos
QS	0.104	0.084
IS	0.360	0.295
Sharpness	0.223	0.182
ACE	3.140	2.151

## 6. CONCLUSIONES

En 2021 la eólica fue la principal fuente de energía eléctrica en España. No obstante, a pesar de su importancia, en la actualidad sigue resultando difícil hacer estimaciones sobre la posible producción futura (a un día vista) debido a la complejidad y a la naturaleza caótica del viento.

Por ello en este trabajo se han realizado predicciones probabilísticas de la generación de energía eólica en la región de Cádiz durante los años 2015-2018 con el fin de tener una cantidad mayor de información a la hora de tomar decisiones respecto a la cantidad a ofertar al día siguiente en la subasta eléctrica.

Para hacer estas predicciones, se han utilizado distintos modelos, comparando sus resultados con distintas métricas de evaluación. El modelo que mejores resultados obtiene parece ser la LSTM utilizando todas las variables de entradas y mirando 6 días hacia atrás.

Además, se ha comprobado que la función de coste Smooth Pinball obtiene mejores resultados que la función de coste Pinball.

Si bien los resultados todavía podrían mejorarse, añadiendo nuevas variables o con modelos algo más complejos, creemos que la LSTM actual podría ser lo suficientemente buena como para testarla en otras regiones de España.