

Herramientas de soporte al trading de commodities

MÁSTER EN BIG DATA Y ANALÍTICA AVANZADA

ELADIO JIMÉNEZ-ESPADAFOR SARDÓN

Contenidos

Introducción.....	4
Trading automático con derechos de emisiones de CO ₂	5
Objetivo y contexto.....	5
Motivación para el desarrollo de un modelo de derechos de emisiones de CO ₂	7
Parque de generación térmica de Iberdrola	7
Consideraciones para la elaboración del modelo y estado del arte	8
Elementos de un problema de RL	11
Procesos de decisión de Markov.....	11
Recompensas	12
Value function	12
Política	13
Model-based vs model-free	13
Desarrollo del modelo	14
Metodología	14
Resultados y conclusiones.....	14
Anexo A – Librerías de RL utilizadas.....	25
Anexo B – Explicación del funcionamiento de los algoritmos utilizados.....	27
Anexo C – Definición de términos e indicadores financieros	29
Bibliografía	29
Previsiones de generación fotovoltaica	30
Introducción.....	30
Motivación.....	30
Objetivos y metodología	30
Contexto de RUR en Iberdrola	31
Datos disponibles.....	31
Exploración de datos y consideraciones previas al modelado	34
Origen de datos	34
Tratamiento de datos	37
Dask - Herramienta de tratamiento de datos	37
Herramientas de modelado	41
DataRobot	41
Metodología	41

Resultados y conclusiones	44
Figura 1 – Fases de implementación del EU ETS. Fuente: EU ETS handbook	6
Figura 2 - Evolución de emisiones de efecto invernadero en la Unión Europea entre 1990 y 2018 (azul) y objetivos previos al Clean Energy Package de 2018. Fuente: European Environment Agency.....	7
Figura 3-Histórico de precios de derechos de emisiones de CO2 2008-04-08 a 2021-04-16. Fuente: quandl.....	8
Figura 4 - Ejemplo del resultado de optimización de una cartera	10
Figura 5 - Esquema de elementos esenciales en un problema de RL	12
Figura 6-Taxonomía de algoritmos de RL. Fuente: spinningup.openai.com.....	13
Figura 7-Serie de posiciones del agente entrenado con 1000 episodios y normalización de las observaciones de estado.....	15
Figura 8-distribución de retornos mensuales.....	16
Figura 9-Volatilidad y Sharpe para 1000 ciclos de entrenamiento con normalización..	16
Figura 10-Retornos mensuales para 1000 muestras sin normalización	17
Figura 11-1000 ciclos de entrenamiento sin normalización	17
Figura 12-Distribución de retornos mensuales para 1000 muestras sin normalización	18
Figura 13-Volatilidad y Sharpe resultados de 1000 muestras sin normalización	18
Figura 14-Retorno del agente sobre diferentes horizontes de tiempo	19
Figura 15-3000 ciclos con normalización	19
Figura 16-Distribución de retornos de agente entrenado con 3000 muestras normalizadas.....	20
Figura 17-distribución de retornos en distintos horizontes temporales.	20
Figura 18-Serie de posiciones del agente entrenado con 3000 muestras sin normalización	21
Figura 19-Distribución de retornos de agente entrenado con 3000 muestras sin normalizar.....	21
Figura 20-Distribución de retornos en distintos horizontes temporales	22
Figura 21-5000 muestras de entrenamiento normalizadas	22
Figura 22-Distribución de retornos mensuales	23
Figura 23-Cuantiles de retorno.....	23
Figura 24-5000 ciclos sin normalización.....	24
Figura 25 - arquitectura de la red neuronal de q-trader	25
Figura 26-Muestra del conjunto de datos Medidas consolidados. Se muestran únicamente datos agregados por motivos de privacidad de datos.	31
Figura 27-Muestra del conjunto de datos Prevision unitaria meteo	31
Figura 28-Produccion fotovoltaica para de una provincia. Se omiten las unidades y la provincia por motivos de protección de datos de clientes de Iberdrola	32
Figura 29- Muestra del conjunto de datos Previsiones agregadas. Se muestran datos agregados por razones de privacidad.....	32

Figura 30-Illustración de los datos contenidos en los ficheros para el dato <<TOTAL1>> y la serie de una provincia. Se omiten la provincia y las unidades por motivos de protección de datos.....	33
Figura 31-Generación fotovoltaica mensual de generadores sujetos a RUR.....	34
Figura 32-Illustración simplificada de un modelo de previsión meteorológica. Fuente: Refinitiv.....	35
Figura 33-Representación gráfica del conjunto de simulaciones de un modelo de conjunto. Fuente: Refinitiv	36
Figura 34-Illustración del método de interpolación bilineal. Fuente: https://es.wikipedia.org/wiki/Archivo:Bilinear_interpolation_visualisation.svg	37
Figura 35-Illustración del proceso de generación de grafos y ejecución en hardware. Fuente: https://docs.dask.org/en/latest/scheduling.html	38
Figura 36-Illustración del dashboard de seguimiento de tareas en una máquina.....	39
Figura 37-Illustración de los dashboards de Dask para el eguimiento de carga de memoria y CPU	39
Figura 38-Illustración de la estructura de un dask dataframe. Fuente: https://docs.dask.org/en/latest/dataframe.html	40
Figura 39-Comparación de popularidad entre varios frameworks de programación distribuida. Fuente: stackoverflow trends	41
Figura 40-Tres mejores modelos por puntuación MAE.....	42
Figura 41-Sensibilidad del modelo seleccionado a los hiperparámetros.....	43
Figura 42-Importancia de las variables de entrada en la predicción del modelo	44

Introducción

Este proyecto se llevó a cabo en Iberdrola. Iberdrola es una corporación global del sector de la energía creada en noviembre de 1992, tras la fusión de Hidroeléctrica española e Iberduero. Es el número uno mundial en producción de electricidad a partir de energía eólica y una de las mayores empresas del mundo de servicios de electricidad por capitalización de mercado, de aproximadamente 73,23 mil millones de dólares.

El grupo es dueño de negocios en numerosos países:

- España
- Reino Unido
- Estados Unidos de América
- México
- Brasil
- Alemania
- Portugal
- Italia
- Francia
- Australia
- Japón
- Grecia
- Irlanda
- Rumanía
- Hungría
- Chipre

Su principal ámbito de actividad se encuentra entorno a los servicios públicos de generación, transporte, distribución y comercialización de electricidad. Los negocios de Iberdrola en España engloban la generación de electricidad, la distribución y comercialización. Este proyecto está relacionado con las actividades de generación y comercialización de electricidad en España, a las que da servicio el departamento de Global Energy Management (GEM), aunando las necesidades de representación en mercado y gestión del riesgo de la cuenta de resultados de ambas actividades de la cadena de valor de la electricidad.

En este contexto, pretendemos encontrar solución a dos problemas distintos que pueden ser resueltos con inteligencia artificial:

- Ajustar previsiones de generación de electricidad fotovoltaica para evitar penalizaciones en la actividad de venta de electricidad como consecuencia de desvíos en la generación real respecto de la prevista.
- Diseñar un robot para aprovisionamiento automático de derechos de emisiones de CO₂, que consiga reducir los costes asociados en la generación de electricidad a partir de combustibles fósiles.

Dado que se trata de dos proyectos independientes se han tratado de esta manera a la hora de realizar esta memoria y ambos apartados pueden leerse por separado.

Aprovisionamiento automático de derechos de emisiones de CO₂

Objetivo y contexto

El trabajo se desarrolló en la mesa de *trading* de Iberdrola. El objetivo de este equipo es diseñar, valorar, negociar y suscribir instrumentos contractuales orientados a las necesidades de los clientes en España, Portugal, Francia, Alemania e Italia o en cualquier otro país con actividad comercial, estructurando los precios de transferencia y cuantificando el riesgo que conlleva la contratación de cada producto para asegurar el margen de beneficio comprometido. Esto incluye la negociación en mercados a plazo de electricidad, emisiones de CO₂, certificados de capacidad eléctrica, petróleo y gas, en mercados organizados, OTC (over-the-counter), subastas y negocios bilaterales con contrapartes.

El objetivo de este proyecto consistió en la elaboración de un modelo que permitiese evaluar estrategias de operación en el mercado de derechos de emisiones de CO₂, minimizando su coste de aprovisionamiento para la compañía. “Un derecho de emisión autoriza al propietario a emitir una tonelada de CO₂ o cantidad equivalente de gas de efecto invernadero. Los estados miembros de la unión expiden nuevos derechos de emisión el 28 de febrero de cada año [...]. Los derechos de emisiones tienen validez de un año. El 30 de abril de cada año, se requiere a las empresas cancelar el número de derechos (EUAs¹) correspondiente a su nivel de emisiones del año anterior”.²

Para Iberdrola, estos costes están asociados a las emisiones de gases de efecto invernadero por centrales eléctricas de ciclo combinado que funcionan por combustión de gas natural.

El Régimen de comercio de derechos de emisión de la UE (EU ETS por sus siglas en inglés) fue creado en 2005 por la Unión Europea. El objetivo es conseguir reducir las emisiones de gases de efecto invernadero dentro de la Unión Europea de la manera más efectiva y económica posible.

Los principales hitos por conseguir del mecanismo son la reducción en un 55% de las emisiones en 2030 respecto de 1990 y que los países europeos sean una economía neutra en emisiones de efecto invernadero en 2050.

El régimen se organiza temporalmente en fases de negociación. Estas fases de negociación se diferencian en modificaciones del marco legislativo que se realizan a partir de las lecciones aprendidas en las fases anteriores, y deben proponerse y aprobarse en el parlamento europeo. Tienen el objetivo de mantener el mercado de

¹ European Union Allowances

² <https://www.nasdaq.com/solutions/eua-carbon-emission-futures-options>

derechos de emisiones en coherencia con su propósito original, adaptándose a cambios económicos, financieros y tecnológicos.

Estas modificaciones han lidiado con el efecto de crisis económicas en el tejido productivo europeo, que han reducido el conjunto de emisiones netas en mayor medida de lo previsto e incurriendo en precios de emisiones bajos, resultando en un sistema poco restrictivo. Una de las principales enmiendas fue la creación de la reserva de estabilidad del mercado (MSR, por sus siglas en inglés). Consiste en una medida de largo plazo que se introdujo en enero de 2019 para abordar problemas de excedentes de derechos de emisión y ajustar la oferta cuando existan perturbaciones importantes en el mercado como consecuencia del entorno macroeconómico.



Figura 1 – Fases de implementación del EU ETS. Fuente: EU ETS handbook

Al ser un mercado artificial, en el sentido de que no se comercia con un recurso natural o un producto físico, sino derechos (que pueden entenderse como licencias), y formando la intervención en el mercado una componente inherente a este, el modelado presenta una problemática muy diversa.

El sistema funciona bajo el principio de *limitar y comerciar (cap-and-trade)*. La comisión pone un límite máximo a la cantidad de gases de efecto invernadero que pueden emitir las instalaciones contempladas en el régimen. Este límite garantiza que los derechos de emisiones tengan un valor³ al convertirse en un recurso limitado y fungible, es decir, puede usarse una sola vez y en un espacio temporal acotado. Estas características convierten los derechos de emisiones en un bien económico comerciable.

El diseño del sistema de derechos de emisiones da flexibilidad a los negocios que incluye para decidir qué tipo de medidas tomar frente a sus emisiones de efecto invernadero. Las actividades contaminantes que tengan costes de reducción de emisiones inferiores al precio de los derechos de emisión tienen un incentivo para modificar sus instalaciones en el sentido de reducir sus emisiones. Los contaminantes con costes asociados a la reducción de emisiones superiores al precio de los derechos de emisión pueden comprarlos y posponer la modificación de sus instalaciones, haciendo frente a sus obligaciones respecto a la política de emisiones de efecto invernadero de una forma más económica.

³ https://ec.europa.eu/clima/policies/ets_es

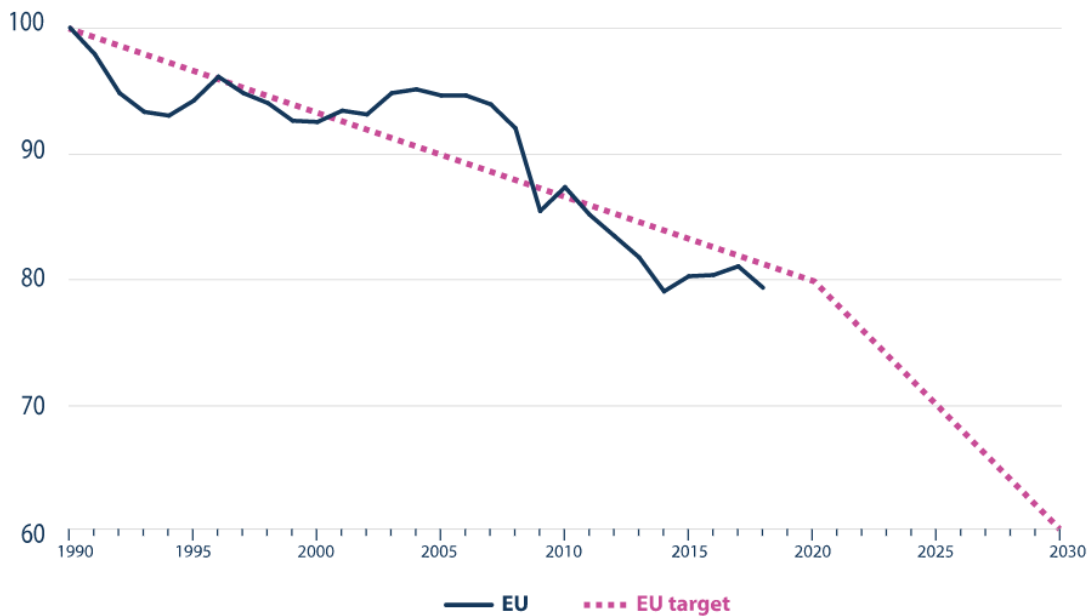


Figura 2 - Evolución de emisiones de efecto invernadero en la Unión Europea entre 1990 y 2018 (azul) y objetivos previos al Clean Energy Package de 2018. Fuente: European Environment Agency.

Motivación para el desarrollo de un modelo de derechos de emisiones de CO₂

Parque de generación térmica de Iberdrola

Iberdrola consta en su parque de generación de electricidad en España con centrales de ciclo combinado que funcionan con gas natural. Como resultado de esta actividad se incurre en emisiones de gases de efecto invernadero. Para la generación de electricidad que se encuentre dentro de las fronteras de la Unión Europea, Iberdrola tiene la obligación de comprar los derechos de emisiones correspondientes a las toneladas emitidas durante cada año natural

La motivación de elaborar estrategias de operación en mercados de CO₂ se justifica por la tendencia alcista de precios de emisiones desde 2015. En este periodo su precio se ha multiplicado por diez, de aproximadamente 5€/t en 2016 a 50€/t en abril de 2021.

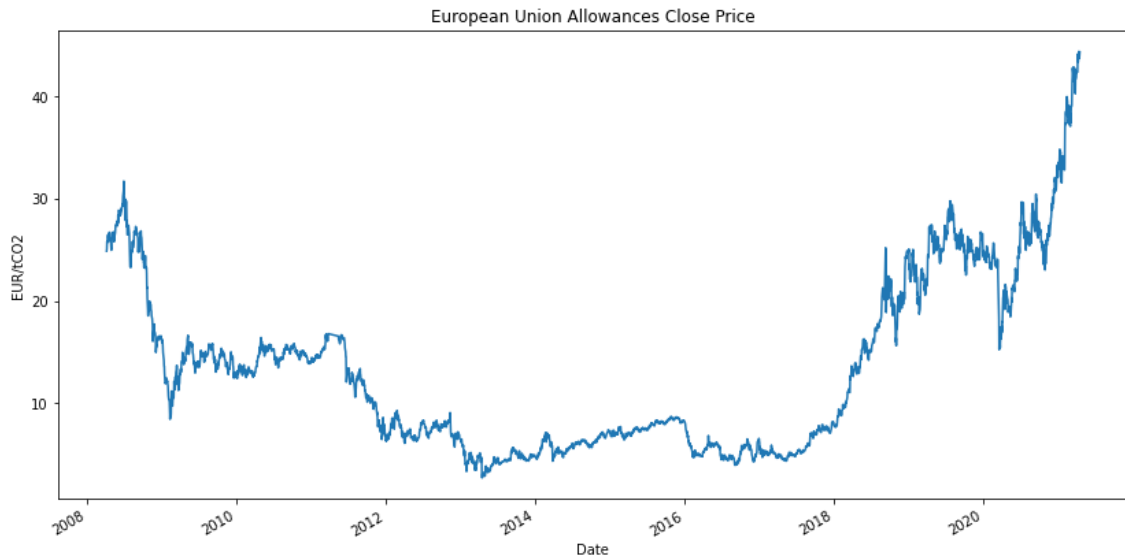


Figura 3-Histórico de precios de derechos de emisiones de CO₂ 2008-04-08 a 2021-04-16. Fuente: *quandl*

Consideraciones para la elaboración del modelo y estado del arte

Existen varios tipos de modelos que pueden ajustarse a las necesidades del proyecto. Hemos considerado tres tipos de modelos que no podíamos eludir:

- Modelos de previsión de precios
- Optimización de portafolios
- Reinforcement Learning

Existe una amplia literatura entorno a proyecciones de precios. En el ámbito financiero están ligadas en su mayoría a las previsiones de precios de cotizaciones bursátiles y tipos de interés. Ante la necesidad de acotar la investigación en la literatura se redujo al tema de previsiones de precios de CO₂. No se incluye documentación relativa a series temporales por haberse estudiado ya en el máster.

En el caso del mercado de emisiones europeo la literatura está concentrada en propuestas estructurales en lugar de métodos puramente estadísticos. Las propuestas estructurales pretenden modelar precios a partir de la simulación del proceso de toma de decisiones. En la bibliografía⁴ se realiza la estimación de precios de EUA a partir de la demanda de electricidad. Al ser la generación de electricidad una actividad intensiva en emisiones de efecto invernadero, los autores del artículo proponen la construcción de un *merit order*⁵ que provee una ordenación de despacho de electricidad a partir de las fuentes de generación de electricidad según sus costes, en un determinado periodo. Este proceso permite estimar las emisiones de CO₂ del sector energético y atribuirles un precio.

En recientes artículos⁶ se propone también un modelo estructural para el modelado de precios. El objetivo de los autores se centra en explicar cuáles son las razones inherentes

⁴ <https://arxiv.org/abs/1011.3736>

⁵ <https://www.cleanenergywire.org/factsheets/setting-power-price-merit-order-effect>

⁶ <https://www.econstor.eu/handle/10419/216726>

de las variaciones de precios que se han observado en el mercado y elaborar posteriormente, en otro artículo, una crítica sobre la idoneidad de la estructura (regulación y participantes) del mercado. Los autores utilizan un modelo teórico como punto de partida para después relacionar tres causas posibles de las variaciones de precios: elementos fundamentales de la demanda, intervenciones regulatorias y finanzas.

Según el tipo de modelo que se decida hacer finalmente, las previsiones de precios pueden ser una componente necesaria para el modelo, pero nunca suficiente. Es un elemento del puzle que ayuda a construir la estrategia, pero no provee de por sí ninguna.

Sería necesario un módulo de toma de decisiones. **En este caso se pensó en optimización de portafolios.** Estos modelos son de uso extenso en finanzas, ayudando a equilibrar la combinación de activos atendiendo a un compromiso entre el retorno y el riesgo de la cartera. La teoría del portafolio moderna⁷ aplicada a una cartera de inversión resulta en un conjunto de activos sobre los que se diversifica el capital de la cartera. Puede parecer contraintuitivo aplicar este método cuando únicamente se está comerciando con derechos de emisiones. En el mercado de emisiones existen realmente varias opciones para compensar las emisiones. Los productos de derechos de emisiones de CO₂ son mensuales, y todos son válidos para la compensación de emisiones de CO₂ al final del año. Estas alternativas presentan la opcionalidad a la que puede atenerse un modelo de optimización de portafolio. Sin embargo, para que el modelo sea efectivo, es necesario que exista una baja correlación entre las opciones, de manera que la tendencia de precios de los distintos productos no sea la misma. Este no es el caso del mercado de CO₂.

⁷ https://en.wikipedia.org/wiki/Modern_portfolio_theory y <https://youtu.be/ywl3pq6yc54>

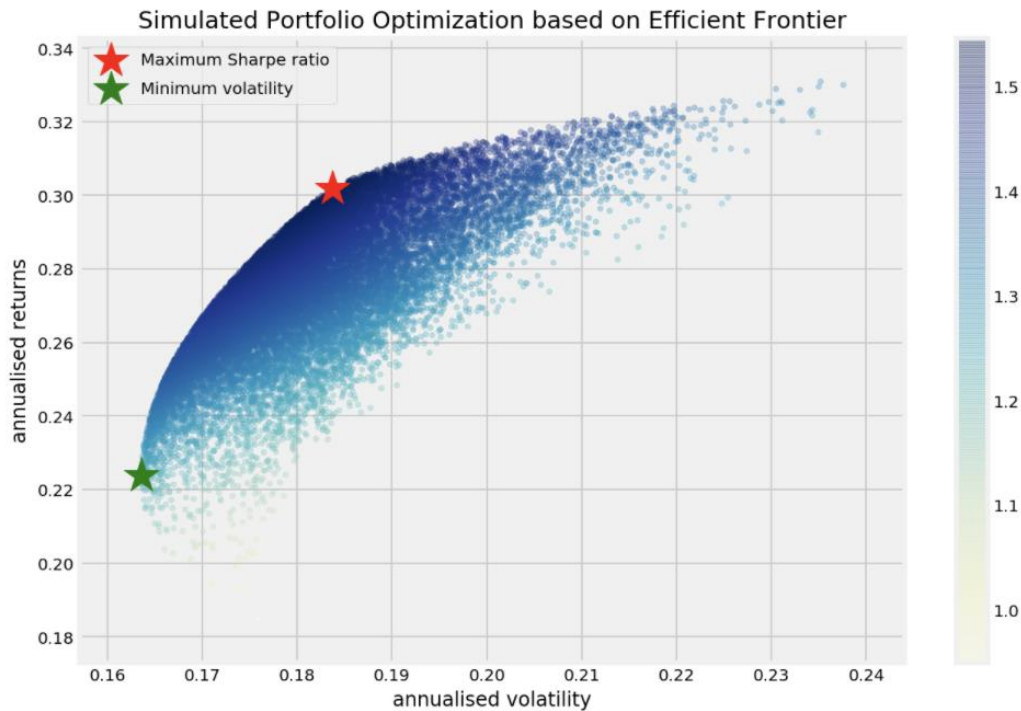


Figura 4 - Ejemplo del resultado de optimización de una cartera

Además, estos modelos suelen requerir una alta capacidad de computación y el conocimiento de técnicas de teoría de control y aproximación a la resolución del problema que quedarían fuera del horizonte del máster.

La literatura en Reinforcement Learning aplicada a operaciones en mercados financieros propone soluciones de aprendizaje automático que nos permiten abstraernos tanto del problema de optimización como del modelado explícito de precios. También tenemos la ventaja de que este tipo de modelos ha sufrido un auge de popularidad reciente y existe una amplia comunidad que contribuye a proyectos de software libre relacionados, facilitando su implementación.

En Reinforcement Learning se entrena un agente orientado a un objetivo determinado y que interactúa con un entorno estocástico. RL es especialmente adecuado para trading automático por la naturaleza interactiva y on-line del problema y la hipótesis de mercado perfecto, que está directamente alineada con la propiedad de Markov⁸. La hipótesis de mercado perfecto, en el caso del CO₂, significa que el precio actual está influenciado únicamente por la información de mercado disponible inmediatamente antes de conocer el dato del precio. Esto puede expresarse en términos de la propiedad de Markov indicando que el precio de los derechos de emisiones de CO₂ en el mercado es un proceso estocástico. Cada precio en la secuencia viene determinado exclusivamente por el precio anterior, es decir, es un proceso sin memoria. Esto significaría que no existen tendencias marcadas exclusivamente por precios anteriores en este mercado. Como se puede intuir esto provee el marco teórico necesario para el modelado, pero no se ajusta de manera perfecta a la realidad. Más adelante veremos

⁸https://es.wikipedia.org/wiki/Propiedad_de_Markov

maneras de aproximar el comportamiento real sin comprometer la calidad de la solución.

La gran diferencia conceptual del RL respecto a los otros métodos considerados es que no se limita a hacer proyecciones a futuro ni se apoya exclusivamente en éstas, sino que toma una visión de principio a fin del proceso de toma de decisiones orientadas a alcanzar el objetivo, integrando acciones y sus consecuencias⁹.

Elementos de un problema de RL

Procesos de decisión de Markov

Los procesos de decisión de Markov (MDP¹⁰ por sus siglas en inglés) proporcionan una abstracción teórica del proceso de interacción entre un decidor y el entorno en el que toma las decisiones. También se puede interpretar como un problema de aprendizaje a través de la interacción con un entorno para alcanzar un objetivo determinado. Un proceso de decisión de Markov consiste en un problema de toma de decisiones secuencial a lo largo de una serie de intervalos temporales que constituyen un episodio.

En cada intervalo, la descripción del entorno describe completamente el proceso, es decir, el estado próximo viene únicamente definido por el estado actual y la acción que se decida tomar sobre él. Esto quiere decir que el proceso no tiene memoria y por lo tanto incorporar información de estados pasados no aporta valor al proceso.

En cada intervalo temporal el decidor o agente recibe una observación (o descripción) de en qué estado se encuentra el entorno y decide una acción que realizar sobre él. El agente recibe una recompensa al transitar de un estado a otro entre intervalos temporales. El objetivo del agente consiste en tomar las decisiones que maximizan la suma de recompensas futuras.

⁹ El desarrollo de los modelos de reinforcement learning ha sido en su mayoría informado para la elaboración del proyecto en:

Reinforcement Learning, Industrial Applications of Intelligent Agents, Phil Winder; O'Reilly
Machine Learning for Algorithmic Trading, Stefan Jansen; Packt

¹⁰ Markov Decisions Process

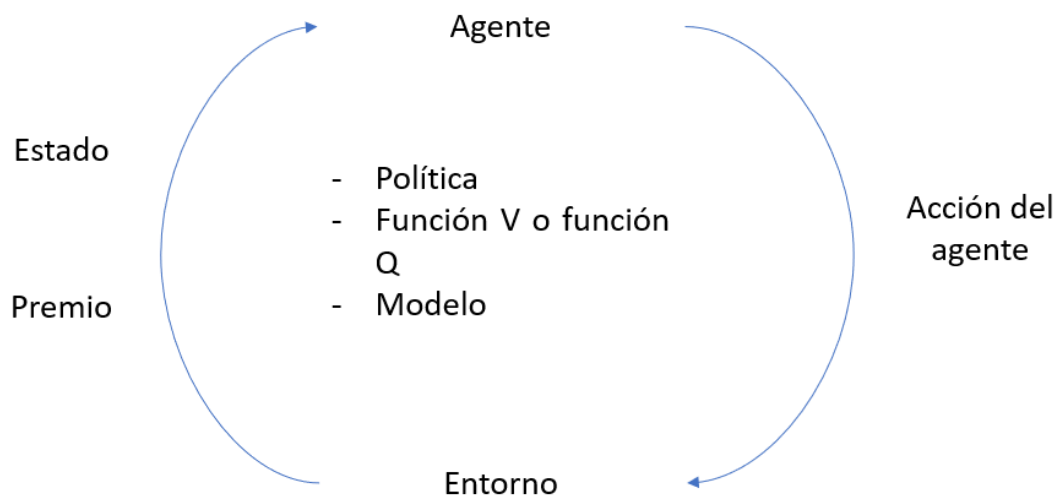


Figura 5 - Esquema de elementos esenciales en un problema de RL

La toma de decisiones del agente se rige por una estrategia o política que pretende escoger las acciones que maximizan la suma de recompensas futuras.

Recompensas

La recompensa es un valor que el entorno envía al agente en cada salto de tiempo. El objetivo del agente es maximizar la suma de recompensas recibidas a lo largo del tiempo. La suma de recompensas suele multiplicarse por un factor de descuento para forzar la convergencia y la pérdida de valor de una recompensa en el tiempo. La suma descontada de recompensas se define como:

$$G_t = R_{t+1} + \gamma R_{t+2} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$R \equiv$ recompensa

$G_t \equiv$ retorno

$\gamma \equiv$ descuento

$t \equiv$ instante de tiempo

La recepción de recompensas del agente es la única manera que tiene de evaluar la optimalidad de su toma de decisiones y modificar la política de manera acorde. El diseño de las recompensas tiene un impacto crítico en el aprendizaje del agente y es una de las tareas más exigentes.

Value function

Para resolver un problema de reinforcement learning es necesaria más información sobre la utilidad de las acciones que se toman que las recompensas, que no son más que información inmediata sobre el resultado de una acción. La *value-function* suple esta necesidad, proporcionando una evaluación de los resultados de las acciones que se toman en el largo plazo.

$$V(s) = \mathbf{E}[G_t | \mathcal{S}_t = s]$$

$V \equiv$ Value function

$\mathcal{S}_t \equiv$ Espacio de estados en instante t

$s \equiv$ Estado actual

Política

La política define la toma de decisiones del agente en todo momento. Relaciona cualquier estado que el agente pueda encontrarse con una o varias acciones.

Cuando el problema presenta un número limitado de estados y acciones, la política puede ser una tabla de correspondencia entre ambos que se completa durante el entrenamiento.

Si se trata de espacios de acciones y/o estados continuos, la política toma la forma de una función que se aproxima con métodos de machine learning.

Model-based vs model-free

Existen dos maneras de abordar el problema cuando consideramos las representaciones del entorno del agente.

En los métodos *model-based* el agente aprende a modelar el entorno para poder planear acciones. Estos modelos pueden usarse para predecir el estado y la recompensa próximos a partir del último estado.

Los métodos *model-free* aprenden heurísticamente, mediante prueba y error.

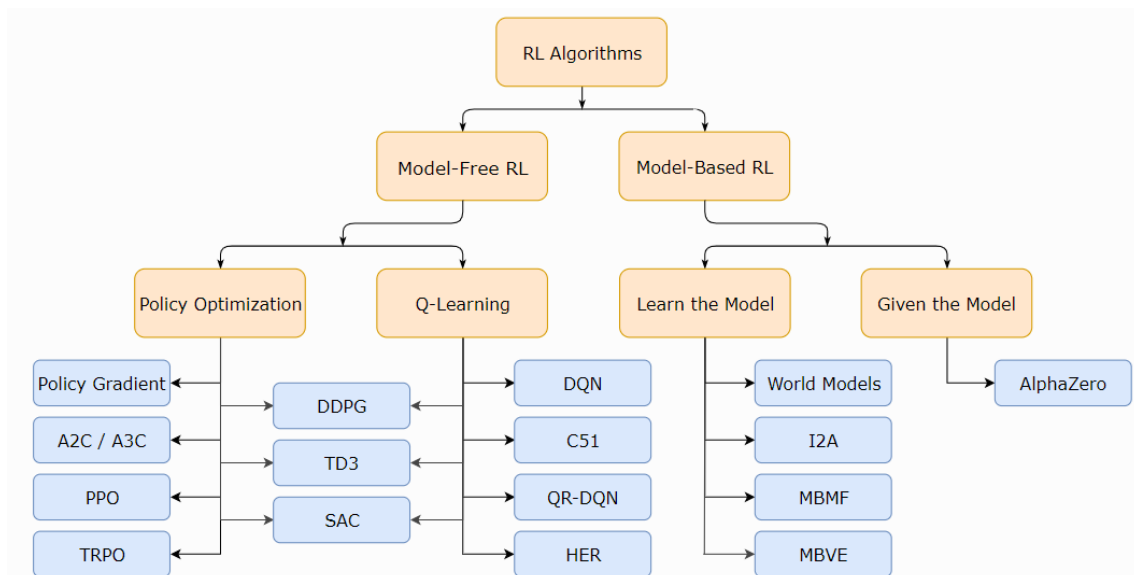


Figura 6-Taxonomía de algoritmos de RL. Fuente: spinningup.openai.com

Desarrollo del modelo

Metodología

Para desarrollar el modelo se ha optado por utilizar librerías de RL disponibles en Python. A continuación, se enumeran las librerías utilizadas:

- Gym
- Gym-anytrading
- Stable Baselines
- Q-trader

Gym-anytrading es una librería que incorpora a los entornos de OpenAI un entorno de trading.

Para más información pueden consultarse los anexos.

Las observaciones de estado que devuelve el entorno son una ventana de 10 precios consecutivos junto con los márgenes obtenidos al realizar las operaciones indicadas por el agente. El objetivo de este agente es obtener estrategias de operación en el corto plazo mediante la identificación de patrones en series de precios de diez días.

No se utilizan períodos de tiempo mayores porque la incertidumbre asociada a los precios de los derechos de emisiones hace que los patrones encontrados en el pasado no sean útiles para elaborar estrategias de manera interactiva.

Al usar gym-anytrading se han utilizado directamente las implementaciones de algoritmos *actor-critic*. En concreto *ACKTR* y *A2C*. Se ha comprobado la efectividad del agente a variaciones en los hiperparámetros del modelo. En concreto al número de muestras usadas para el entrenamiento y normalización de los datos a la entrada de las redes neuronales.

Para cómo de efectivo es el agente se han empleado indicadores financieros:

- Distribución de retornos para diferentes horizontes temporales
- Volatilidad del valor de los activos
-

Estos agentes forman parte de la librería Stable Baselines. En los anexos se proporciona una descripción concisa de la librería comentando sus atributos destacables y se proporcionan referencias tanto a la documentación de la librería como a los artículos donde se detallan los algoritmos.

Resultados y conclusiones

Cómo interpretar las gráficas

Las figuras que siguen ilustran los resultados obtenidos tras el entrenamiento de los agentes para una selección de hiperparámetros determinada.

Las figuras tituladas <<series de posiciones>> muestran la estrategia adquirida por el agente día a día. El agente puede tomar dos estrategias:

- posiciones cortas: El agente decide vender derechos de emisiones con el objetivo de volver a comprarlas a menor precio. En los gráficos los puntos están marcados en rojo. Cuando el agente decide ejecutar esta estrategia no se deberían comprar derechos de emisiones en el trading desk.
- posiciones largas: El agente decide comprar derechos de emisiones ante la expectativa de que el precio aumente en el corto plazo. En la gráfica los puntos están marcados en verde. Cuando el agente decide ejecutar esta estrategia deberían comprarse derechos de emisiones en el trading desk.

Los gráficos de distribución ilustran el acierto de la toma de decisiones del agente. La red neuronal que implementa la política de toma de decisiones proporciona como resultado una distribución de probabilidad. Esto quiere decir que la toma de decisiones es un proceso estocástico. Para comprender el sesgo que existe en la toma de decisiones es necesario observar la distribución de retornos. Así podremos evaluar si la política necesita correcciones, para lo que habría que extender el entrenamiento del agente.

Resultados de 1000 ciclos de entrenamiento con normalización de datos

Total Reward: 1.870000 ~ Total Profit: 0.598375



Figura 7-Serie de posiciones del agente entrenado con 1000 episodios y normalización de las observaciones de estado

Este agente tiene un sesgo claro hacia posiciones cortas. No existe evolución en la toma de decisiones a lo largo del periodo.

Distribution of Monthly Returns

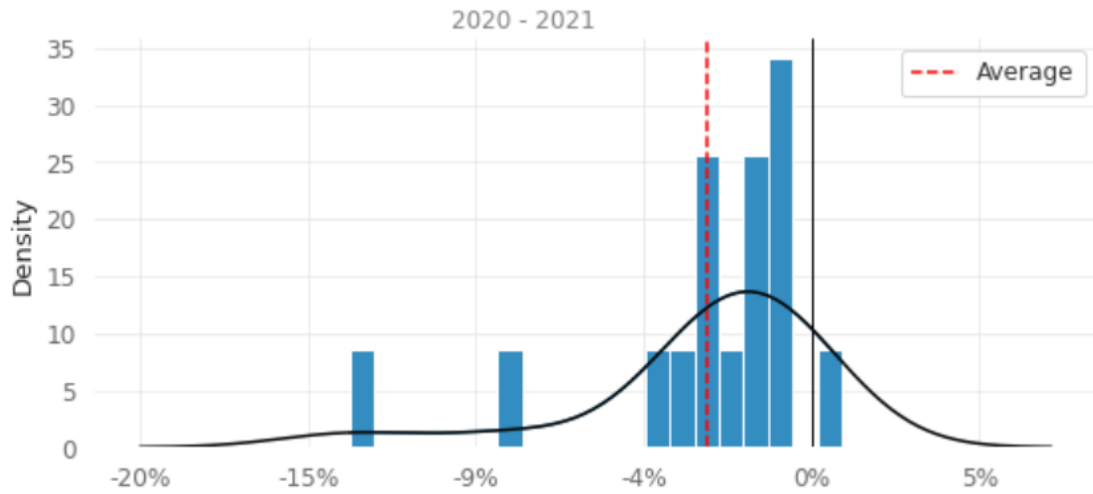
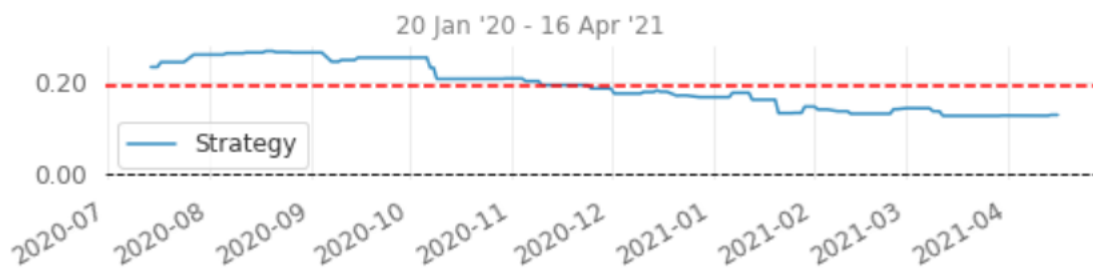


Figura 8-distribución de retornos mensuales

Rolling Volatility (6-Months)



Rolling Sharpe (6-Months)

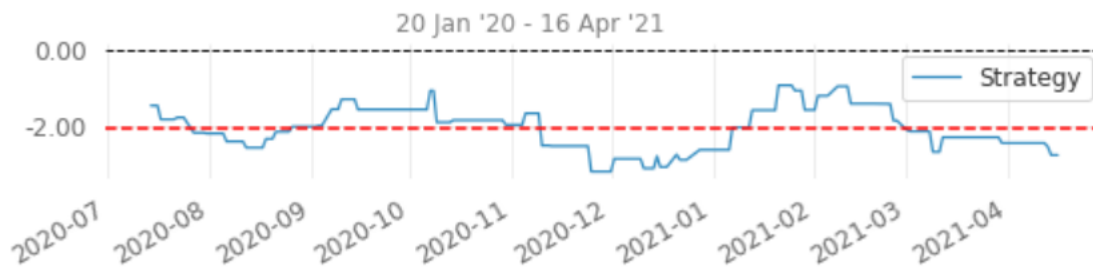


Figura 9-Volatilidad y Sharpe para 1000 ciclos de entrenamiento con normalización

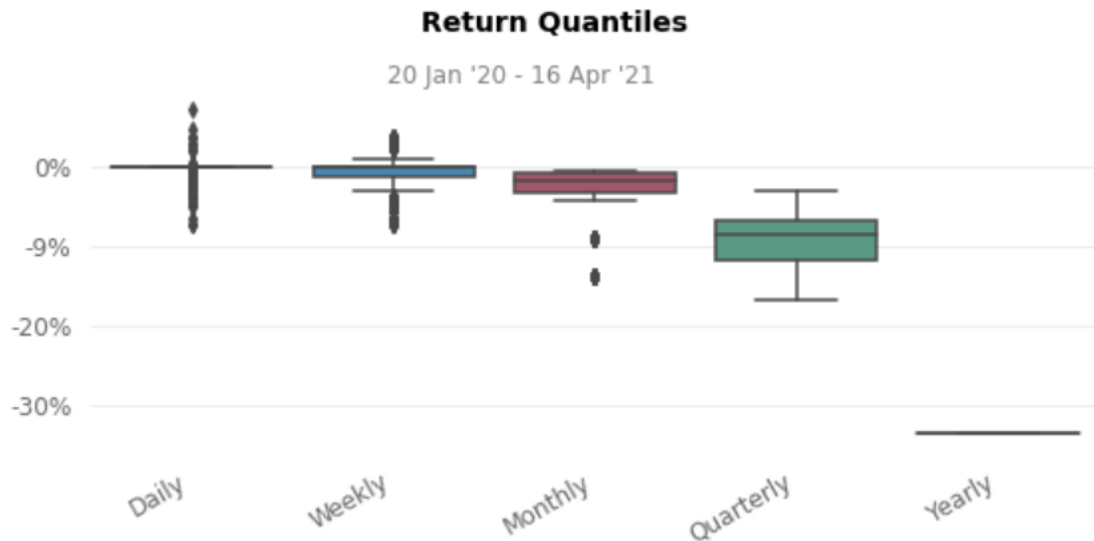


Figura 10-Retornos mensuales para 1000 muestras sin normalización

Resultados de 1000 ciclos de entrenamiento sin normalización de datos

Total Reward: 14.520000 ~ Total Profit: 1.353833

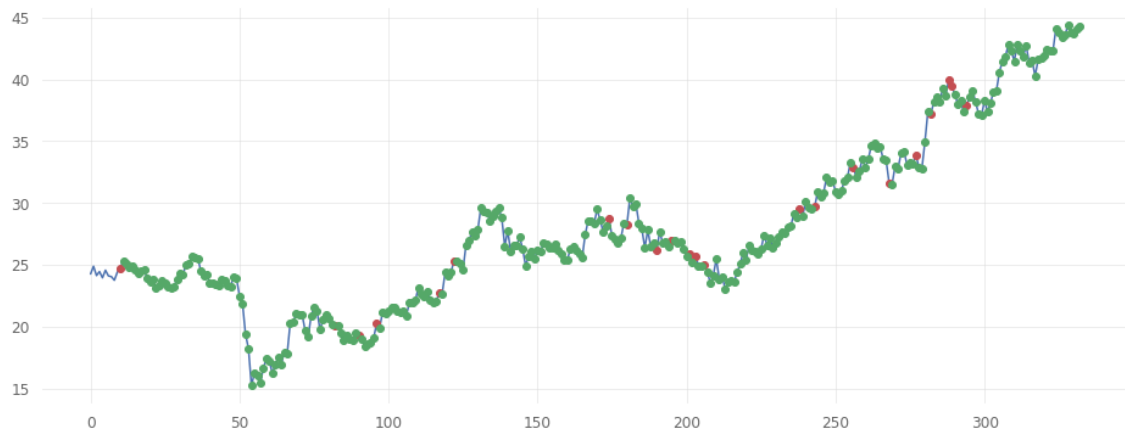


Figura 11-1000 ciclos de entrenamiento sin normalización

Se observa que al no estar los datos normalizados el agente da preferencia a la estrategia de posiciones largas. Se puede observar que al inicio de la serie el agente no toma ninguna posición. Esto se debe a que los primeros diez precios de la serie representan únicamente la primera observación del entorno. Son el estado inicial.

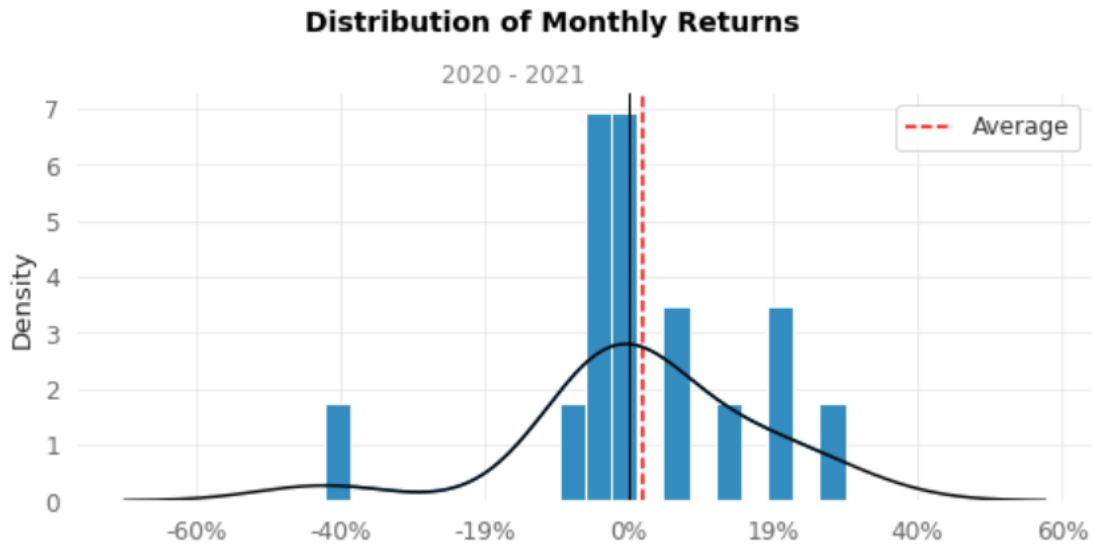


Figura 12-Distribución de retornos mensuales para 1000 muestras sin normalización

La media de los retornos se encuentra sesgada a la derecha y el agente produce retornos muy altos. Esto se debe en gran parte a la evolución alcista de los precios.

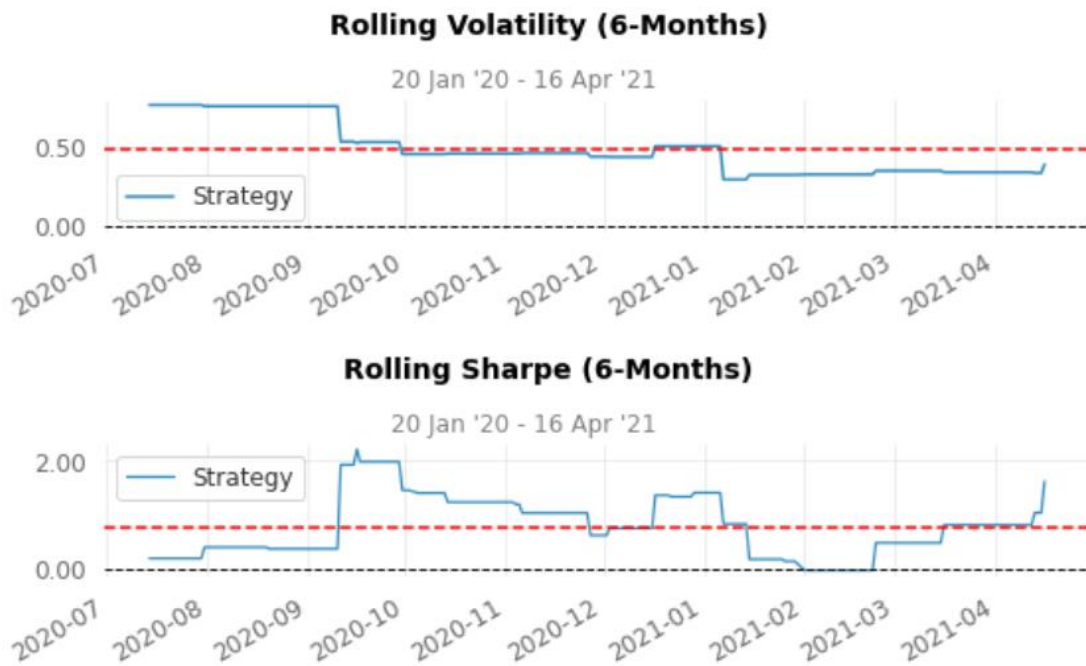


Figura 13-Volatilidad y Sharpe resultados de 1000 muestras sin normalización



Figura 14-Retorno del agente sobre diferentes horizontes de tiempo

Los retornos en el corto plazo nos informan de la capacidad del agente para ejecutar la tarea para la que se ha diseñado. Observamos en la imagen superior que a pesar de los valores atípicos los retornos se encuentran muy centrados. Teniendo en cuenta la alta preferencia del agente para tomar posiciones largas esto refleja sobre todo el carácter estocástico de los precios. En el largo plazo proporciona retornos positivos debido a la tendencia alcista, que hace que la estrategia de posiciones largas sea ganadora.

Resultados de 3000 ciclos de entrenamiento con normalización de datos

Total Reward: 17.900000 ~ Total Profit: 0.553426

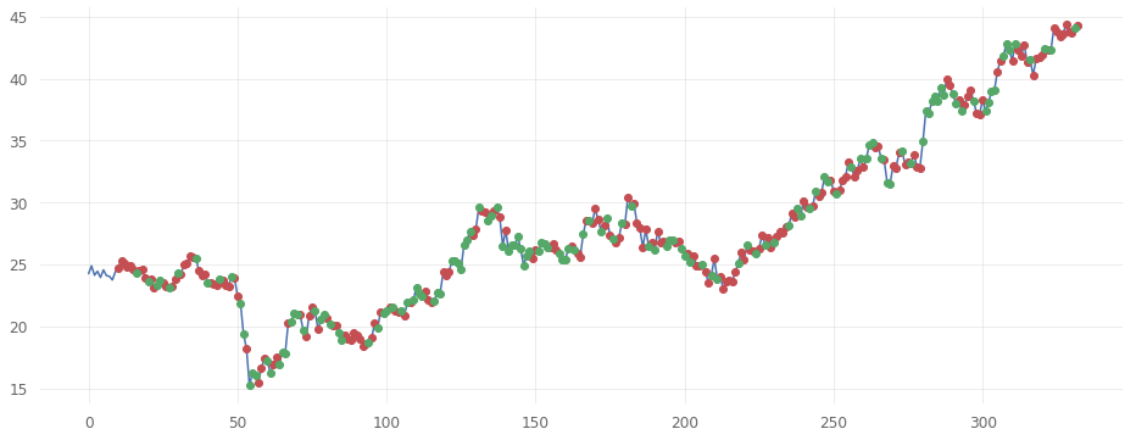


Figura 15-3000 ciclos con normalización

Al entrenar al agente con 3000 muestras no normalizadas la estrategia está menos sesgada hacia posiciones cortas. Aunque esta

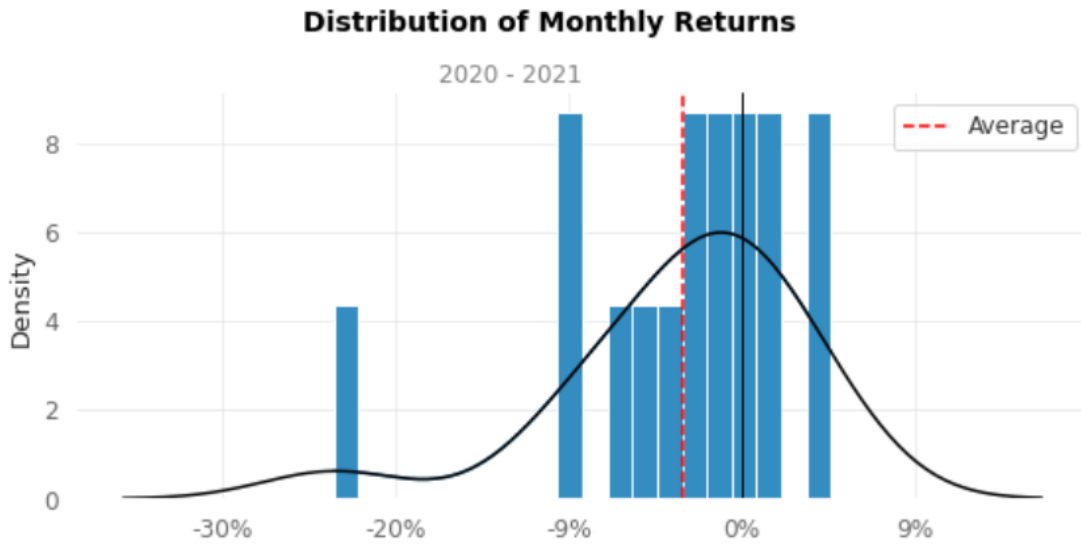


Figura 16-Distribución de retornos de agente entrenado con 3000 muestras normalizadas

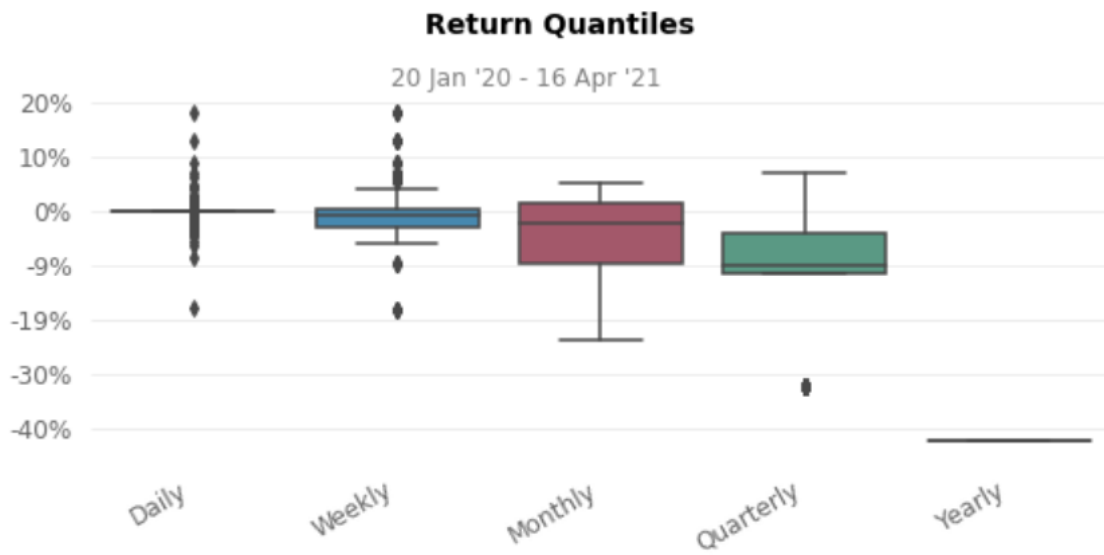


Figura 17-distribución de retornos en distintos horizontes temporales.

Resultados de 3000 ciclos de entrenamiento sin normalización de datos

Total Reward: 17.590000 ~ Total Profit: 1.046313

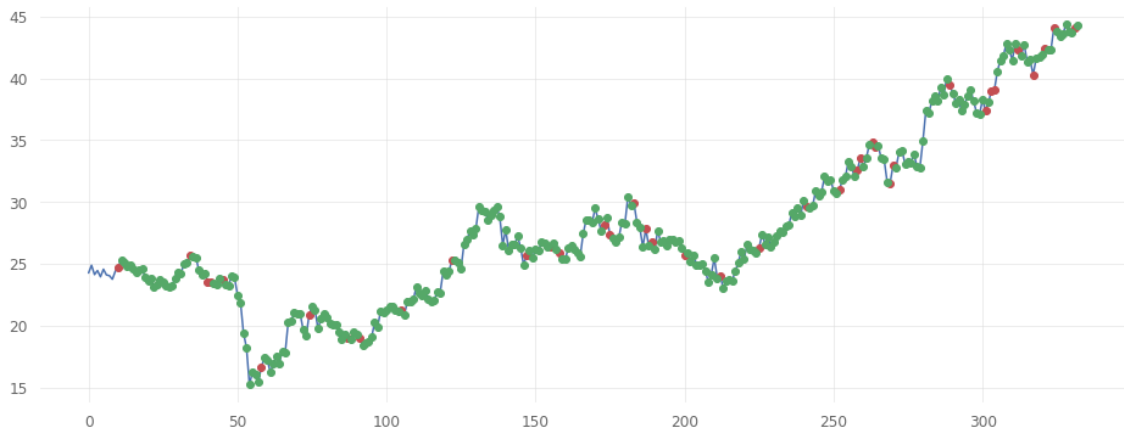


Figura 18-Serie de posiciones del agente entrenado con 3000 muestras sin normalización

Se observa que la preferencia del agente por posiciones sigue siendo similar al caso de 1000 muestras de entrenamiento sin normalización.

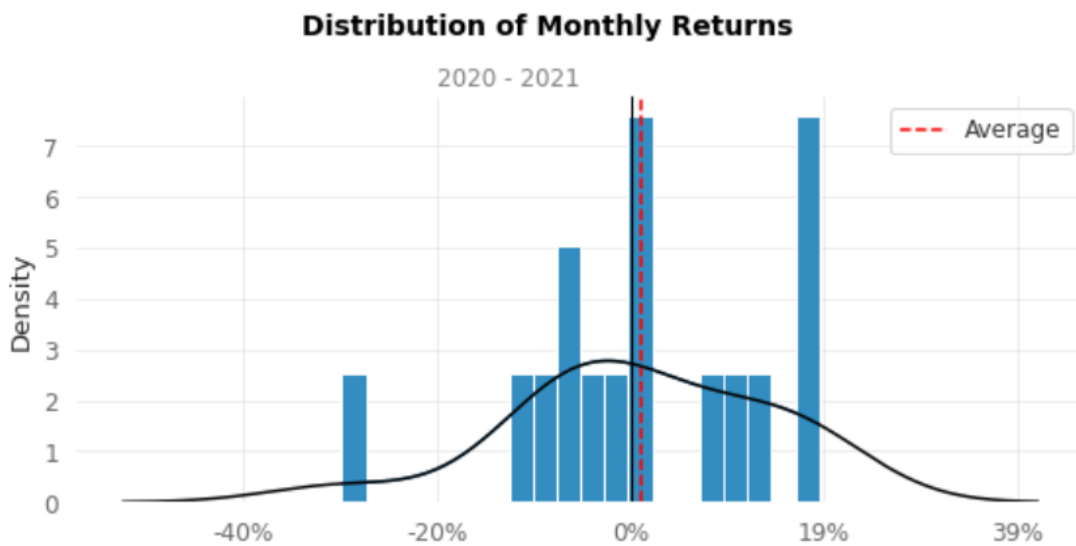


Figura 19-Distribución de retornos de agente entrenado con 3000 muestras sin normalizar

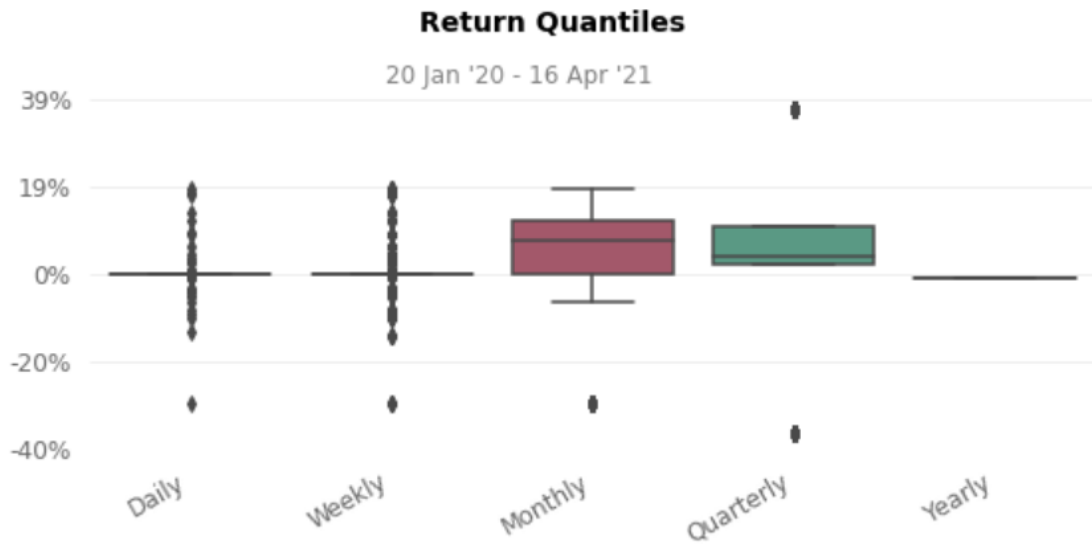


Figura 20-Distribución de retornos en distintos horizontes temporales

Los retornos según horizontes temporales muestran una estructura similar al caso de 1000 muestras. Hay que remarcar que el resultado mensual ha mejorado. Los outliers negativos se encuentran más próximos a la medida central y el tercer cuartil se sitúa por encima del 0%.

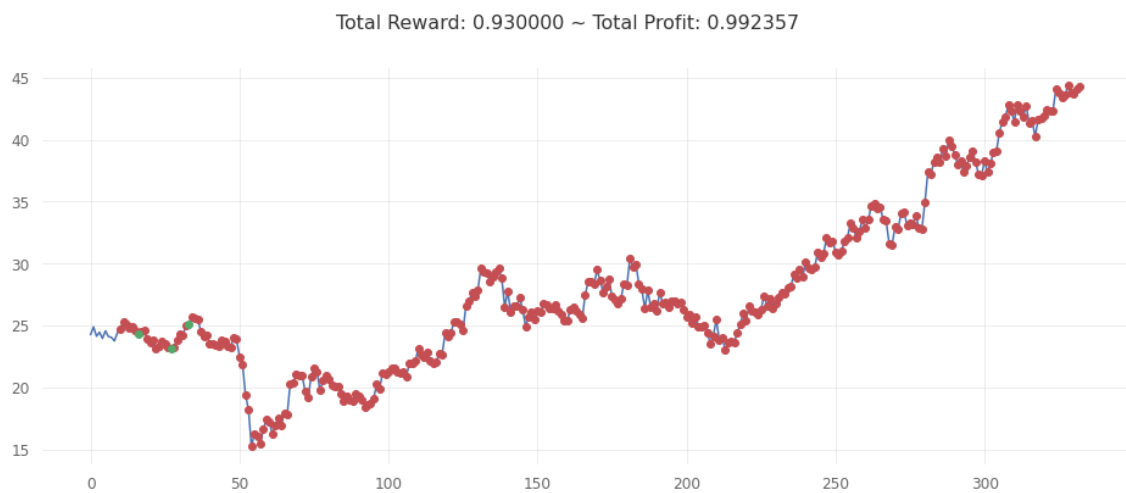


Figura 21-5000 muestras de entrenamiento normalizadas

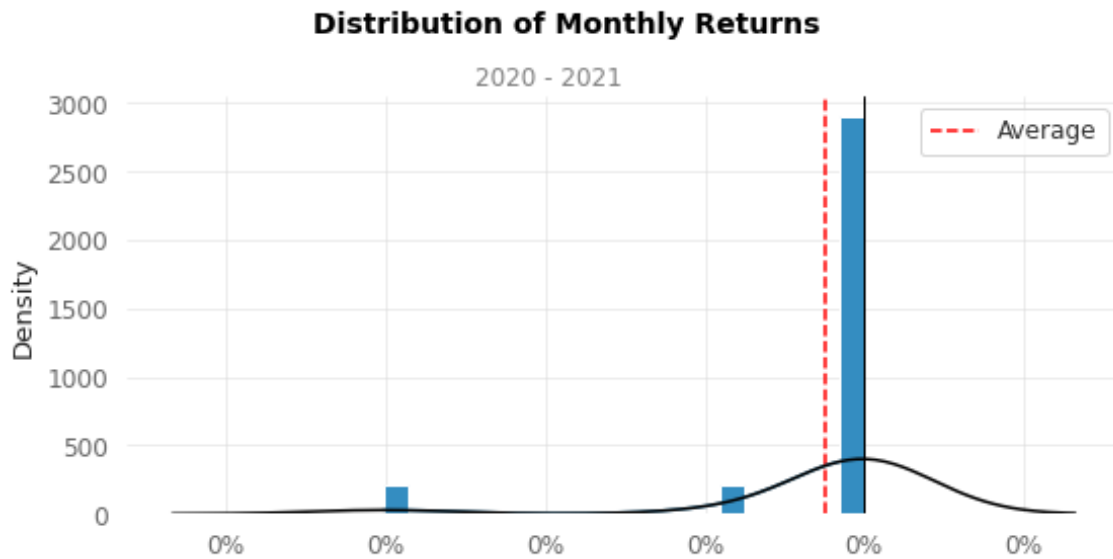


Figura 22-Distribución de retornos mensuales

El entrenamiento con datos normalizados ha provocado que el agente únicamente tome posiciones cortas. Podemos concluir que normalizar la observación del estado a la entrada del agente no produce resultados razonables. Al modificar el orden de magnitud de los datos de entrada los patrones que la red aprende no se corresponden con las magnitudes reales del mercado.

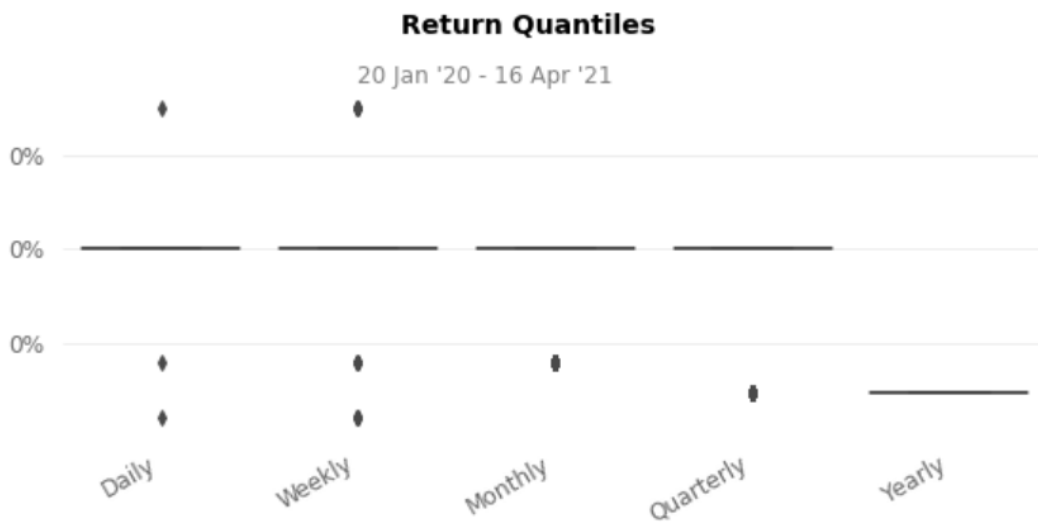


Figura 23-Cuantiles de retorno



Figura 24-5000 ciclos sin normalización

Las conclusiones son:

- Los agentes necesitan muchos ciclos de entrenamiento para alcanzar resultados satisfactorios. Al menos más ciclos de los que permite el hardware de que se dispuso. El aprendizaje basado en policy gradients es más lento que el basado en q-valores. La recomendación de uso de q-trader era de realizar al menos 200 episodios de entrenamiento, y en este caso el agente estaba basado en q valores. Esto requeriría unas 200 horas de entrenamiento. Entrenando los agentes con 10000 episodios utilizando la librería Stable Baselines en Google Colab el servidor mata el proceso.
- En la situación alcista en la que se encuentra el mercado de derechos de emisiones desde 2015 la estrategia a posteriori más lógica consistiría en comprar cuanto antes los derechos de emisiones necesarios para cumplir con las obligaciones de emisiones, algo comúnmente denominado “banking”¹¹, incluso pudiendo rolar los derechos acumulados año a año.
- Sería de mucha ayuda poder realizar un modelo que aproxime el comportamiento de los participantes del mercado, en un modo parecido a los modelos de equilibrio de oferta y demanda de los sectores de la electricidad y el gas. Esto podría proveer una señal de precios que se anticipase al mercado. Sin embargo, esta tarea sería un proyecto aparte y queda fuera del alcance de este. La falta de transparencia en el mercado y la aparente utilidad de los derechos de emisiones de CO₂ como activo financiero requeriría un modelo basado en datos financieros y macroeconómicos.

¹¹ <https://www.emissions-euets.com/banking>

- La librería Q – trader puede proporcionar resultados satisfactorios si se dispone del hardware apropiado siempre cuando se quiera un agente que opere en el corto plazo. El número mínimo de episodios para entrenar un agente de este tipo es 200. En Google Colab esto duraría aproximadamente 200 horas. No se han expuesto resultados de Q – trader en esta memoria por no ser concluyentes.
- Las aplicaciones de Reinforcement Learning disponibles en software libre están orientadas a la robótica y a resolver partidas de videojuegos clásicos. Aunque el RL tenga otras aplicaciones en entornos industriales y digitales es difícil encontrar librerías bien documentadas y con implementaciones de la misma calidad que las de OpenAI o Google aplicadas al trading. Las aplicaciones al trading que se han encontrado. Aunque las librerías existentes se pueden adaptar la documentación disponible es escasa y el código no está comentado. Esto hace que la adaptación sea muy laboriosa y cometer errores es frecuente.

Anexo A – Librerías de RL utilizadas

En este apartado se provee una breve descripción de las librerías de reinforcement learning que se han utilizado.

Q-Trader

Fuente: [edwardhdu/q-trader: Deep Q-learning driven stock trader bot \(github.com\)](https://github.com/edwardhdu/q-trader)

Consiste en una implementación de Q-learning aplicada a la negociación de *stocks* en el corto plazo. El modelo utiliza ventanas temporales con resolución diaria de precios de cierre de mercado para determinar cuál es la mejor acción a tomar en cada etapa (estado):

- Comprar
- Vender
- Mantener la posición

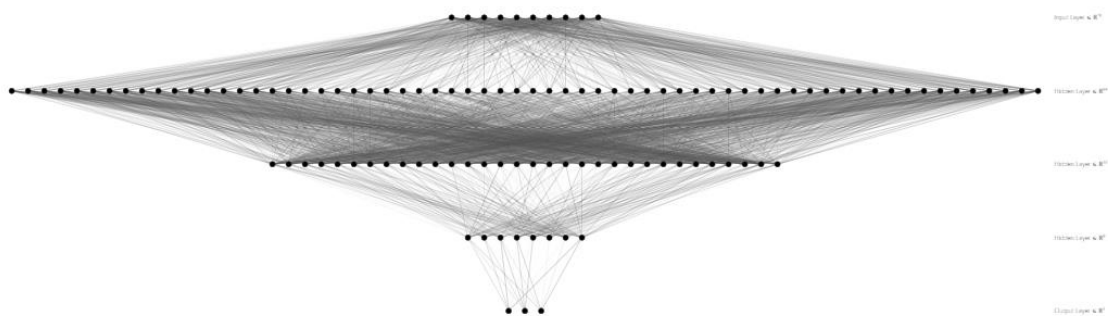


Figura 25 - arquitectura de la red neuronal de q-trader

Como consecuencia de que el estado representa una secuencia de precios en el corto plazo el modelo no da buenos resultados sobre tendencias de largo plazo, pero los resultados son razonables a la hora de predecir picos y valles.

La aproximación de la función acción-valor se realiza mediante una red de cuatro capas completamente conectadas que tiene como salida una capa de tres neuronas. Cada una de las neuronas es una estimación del valor de cada una de las acciones posibles. La neurona de mayor valor indica la acción a tomar.

Esta librería requiere al menos 200 episodios de entrenamiento para que los resultados del agente sean razonables, según las recomendaciones del autor. Un episodio de entrenamiento, es decir, entrenar la red una única vez con los datos de que se dispone, sobre un año, requiere un tiempo de entrenamiento de unos 5 minutos en Google Colaboratory. Esto quiere decir que unas 16 horas son necesarias para el entrenamiento de un único agente. Para el desarrollo de este proyecto se exploró esta librería por la sencillez del código que facilitaba la identificación de los elementos del algoritmo de RL, pero los resultados del agente fueron insatisfactorios ya que no se disponía de 16 horas de uso de los servidores.

Stable Baselines

Fuente: <https://stable-baselines.readthedocs.io/en/master/>

Consiste en un conjunto de implementaciones de algoritmos de Reinforcement Learning basado en [Baselines](#) de [OpenAI](#).

Esta librería presenta una interfaz estándar para el entrenamiento, evaluación y calibración de hiperparámetros de los agentes de RL. La ventaja de esta librería frente a la de OpenAI es que el código, aun siendo muy parecido, está mejor documentado, comentado y el estilo de codificación de toda la librería es parecido. Esto hace más fácil encontrar errores e incompatibilidades.

Al igual que el software libre de OpenAI en RL este paquete presenta soluciones orientadas a la robótica y la solución de juegos de Atari. Esto hace que en ocasiones haya que adaptar elementos de agentes al propósito que se le quiera dar, que puede resultar laborioso y con tendencia a cometer errores e incompatibilidades con otras funciones.

En la siguiente tabla se enumeran los algoritmos que incluye la librería con la bibliografía de referencia en la que se basa la implementación.

Algoritmo	Referencia
A2C	https://arxiv.org/abs/1602.01783
ACER	https://arxiv.org/abs/1611.01224
ACKTR	https://arxiv.org/abs/1708.05144
DDPG	https://arxiv.org/abs/1509.02971
DQN	https://arxiv.org/abs/1312.5602
GAIL	https://arxiv.org/abs/1606.03476
HER	https://arxiv.org/abs/1707.01495
PPO1	https://arxiv.org/abs/1707.06347
PPO2	https://arxiv.org/abs/1707.06347
SAC	https://arxiv.org/abs/1801.01290
TD3	https://arxiv.org/abs/1802.09477

TRPO	https://arxiv.org/abs/1502.05477
Policy Networks	https://stable-baselines.readthedocs.io/en/master/guide/custom_policy.html#custom-policy

Gym-anytrading

Fuente: <https://github.com/AminHP/gym-anytrading>

Esta librería implementa un entorno de entrenamiento de RL a partir de la librería [gym](#) de OpenAI. Proporciona tres entornos de trading:

- Trading Env: Es una clase abstracta de Python que se utiliza como base para implementar las otras dos clases concretas.
- Stocks Env: Es un entorno que simula la operación en mercados de valores
- ForexEnv: Es un entorno que simula la operación en mercados de divisas.

La gran ventaja de este entorno reside en la posibilidad de utilizarlo con la librería [stable-baselines](#), además de ser rápido en el entrenamiento. Es una implementación muy eficiente que ha posibilitado el desarrollo del modelo. También es fácil de usar con otras librerías de análisis financiero como [quantstats](#).

Anexo B – Explicación del funcionamiento de los algoritmos utilizados

En este apartado se da una descripción de los algoritmos de Reinforcement Learning utilizados. Al consistir todos en combinaciones de Q-Learning y Policy Gradients se procede a explicar de manera progresiva los elementos más simples y a la vez generales de cada algoritmo para aportar la teoría fundacional sobre la que se van a construir y explicar las implementaciones de RL que se han utilizado

Q-Learning

Q-learning pretende entrenar un agente para aprender una política que maximice el retorno que recibe del entorno como consecuencia de las acciones que toma. Este método es off-policy porque aprende de acciones que se encuentran fuera de la política actual, como tomar acciones aleatorias, en cuyo caso una política no es necesaria.

El prefijo <<Q>> significa calidad. El algoritmo evalúa la utilidad de los estados en que se encuentra el agente para conseguir una recompensa futura.

Q → quality of a state

Los algoritmos basados en valor aprenden a seleccionar acciones basados en valores estimados de acciones posibles.

Policy Gradient

Los agentes basados en políticas aprenden políticas directamente, relacionando estados de entrada con acciones de salida. Las principales desventajas que presentan frente a los modelos basados en valor son:

- Eficiencia de muestreo: las funciones que aproximan la política requieren realizar el entrenamiento sobre un gran número de muestras de observaciones del entorno para provocar cambios en la política.
- Convergencia: Es frecuente que la optimización de los parámetros de la política lleve a máximos locales, pero no converja a la mejor solución global.

Algoritmos Advantage Actor Critic¹²

Los algoritmos actor critic combinan los dos tipos de algoritmos explicados anteriormente. Estos métodos solucionan la inestabilidad de las políticas y aceleran el entrenamiento. En lugar de esperar al final de cada episodio para actualizar la política, que es cuando se tiene el valor acumulado de todas las recompensas futuras, se utiliza un aproximador del valor de cada estado que sirve para evaluar la bondad de la decisión tomada en cada etapa y actualizar la política. Es un algoritmo *on-policy*.

Estos algoritmos constan de dos redes neuronales que trabajan juntas para resolver un problema. La advantage function calcula el error del agente en el ejercicio de estimación de los valores Q.

La *actor-network* escoge una acción en cada intervalo de tiempo y la *critic-network* evalúa el valor Q de un estado de entrada determinado. A medida que la *critic-network* aprende qué estados proporcionan mayores valores la *actor-network* utiliza esta información para que el agente aprenda a seleccionar acciones que lo sitúen en la trayectoria de estados de mayor valor y lo aparten de estados de menor valor.

Los agentes de aprendizaje por diferencia temporal calibran la toma de decisiones haciendo predicciones de futuras recompensas para las acciones que toman y las ajustan basándose en el error de predicción, que comprueban al finalizar el episodio y recibir la suma descontada de recompensas. La ventaja (*advantage*), se corresponde con el error de predicción del valor de cada estado. Es el error de predicción del agente. Utilizar este método estabiliza el modelo, reduciendo la varianza de la política.

La *actor-network* relaciona cada estado con una acción. Controla el comportamiento del agente. Esta acción será la que maximice la suma de recompensas futuras. La *actor-network* provee como resultado una distribución de probabilidad, teniendo cada acción una probabilidad para cada acción. A partir de la ventaja, durante el entrenamiento, la red calibra sus pesos para que las distribuciones de probabilidad resultantes hagan más probable tomar las mejores acciones.

¹² <https://towardsdatascience.com/advantage-actor-critic-tutorial-mina2c-7a3249962fc8>

La *critic-network* relaciona cada estado con su Q-valor correspondiente. Esta red consiste en una regresión entre el estado de entrada y la estimación de su Q-valor. Cuanto más próxima sea la ventaja a cero, mejor estimación estará haciendo la *critic-network* y mejores decisiones tomará la *actor-network*. Proporciona una medida de cómo de buena es la acción que se ha tomado.

Anexo C – Definición de términos e indicadores financieros

A continuación, se definen los indicadores financieros que se han utilizado para evaluar y comprender las razones del rendimiento del agente de trading.

- Volatilidad (volatility): es una medida estadística de la dispersión de los retornos que proporciona el agente. Cuanto mayor sea la volatilidad, mayor será el riesgo. Se suele medir con la varianza o la desviación estándar.
- Sharpe ratio: Su utilidad es indicar el retorno de una inversión respecto a su riesgo. El ratio se calcula como el retorno medio en un periodo de tiempo por unidad de volatilidad sobre un cierto umbral.

$$\text{Sharpe Ratio} = \frac{R_p - R_f}{\sigma_p}$$

where:

R_p = return of portfolio

R_f = risk-free rate

σ_p = standard deviation of the portfolio's excess return

Bibliografía

- Barto, R. S. (2015). *Reinforcement Learning: An Introduction*. The MIT Press.
- Friedrich, M., Mauer, E.-M., Pahle, M., & Tietjen, O. (2020). From fundamentals to financial assets: the evolution of understanding price formation in the EU ETS. *EconStor*.
- Jansen, S. (2020). *Machine Learning for Algorithmic Trading*. Packt Publishing.
- Schwarz, S. H. (2015). Risk-Neutral Pricing of Financial Instruments in Emission Markets: A Structural Approach, arXiv:1011.3736v3. *arXiv*.
- Winder, P. (2020). *Reinforcement Learning: Industrial Application of Intelligent Agents*. O'Reilly Media.

Previsiones de generación fotovoltaica

Introducción

Motivación

La legislación vigente en España en materia de puesta en marcha del suministro eléctrico de último recurso en el sector de la energía eléctrica, Real Decreto 485/2009, de 3 de abril, regula la comercialización de la producción energética de las instalaciones en Régimen Especial que se hayan acogido a percibir una tarifa regulada, única para todos los periodos de programación, a no ser que los titulares nominales de la instalación decidan y comuniquen la operación a través de otro representante.

Esta regulación obliga a Iberdrola a incluir en sus actividades de operaciones de mercados energéticos estas unidades.

La capacidad de anticipar la generación de estas centrales puede ser un recurso muy beneficioso para evitar, tanto para Iberdrola como para los titulares de las instalaciones, sobrecostes con razón de penalizaciones asociadas a desvíos en la generación incurrida respecto de la declarada.

Las obligaciones contractuales de estos generadores de potencia eléctrica con la entidad interna que les da servicio desde Iberdrola hacen que solamente las unidades de generación que superen 1MW de potencia instalada tienen que enviar sus previsiones de funcionamiento a la entidad.

Los distintos tipos de generadores involucrados requieren diversas metodologías para la previsión de la generación de electricidad. Dado el reducido tiempo disponible para elaborar el proyecto y el acceso disponible a sus recursos nos decantamos por realizar previsiones de generación fotovoltaica a partir de previsiones de incidencia de radiación e históricos de generación fotovoltaica.

Objetivos y metodología

El objetivo del proyecto consiste en elaborar una metodología para realizar las previsiones de generación fotovoltaica.

Esto ha requerido el uso de DataRobot, una herramienta de AutoML con capacidad para entrenar múltiples modelos estadísticos de diversa índole.

Una parte considerable del tiempo dedicado a este proyecto ha sido el tratamiento de datos brutos para generar tablas de datos coherentes en un formato adecuado para la herramienta de AutoML. Debido al gran número de ficheros necesarios para la

elaboración de los datos se han requerido herramientas de paralelización de tareas que reduzcan el tiempo de procesado. La librería de Python Dask ha sido vital en el desarrollo de estos métodos

Contexto de RUR en Iberdrola

Tipos de generadores

Potencias y generación por tipo y año.

Datos disponibles

El inicio del proyecto presentaba la posibilidad de usar hasta cuatro conjuntos de datos independientes para elaborar las previsiones de generación de energía eléctrica a partir de fuentes fotovoltaicas. Estos datos se presentan a continuación:

- <<Medidas consolidadas>>:
Estos datos se corresponden con mediciones de generación de electricidad realizada y potencia disponible en las instalaciones por rango de potencia y provincia de España en la que Iberdrola ejerce de generador de último recurso.

	AÑO	MES	DIA	HORA	RANGO POTENCIAS	COMBUSTIBLE	PROVINCIA	INSTALACIONES	POTENCIA	ENERGIA
0	2017	1	1	1	Menos de 15 Kw	EOLICA	ALBACETE	12.0	72.000000	0.0
1	2017	1	1	1	Menos de 15 Kw	FOTOVOLTAICA	ALAVA	160.0	848.635000	0.0
2	2017	1	1	1	Menos de 15 Kw	FOTOVOLTAICA	ALBACETE	209.5	1714.274999	0.0
3	2017	1	1	1	Menos de 15 Kw	FOTOVOLTAICA	ALICANTE	231.5	1368.180500	0.0
4	2017	1	1	1	Menos de 15 Kw	FOTOVOLTAICA	AVILA	17.5	110.950000	0.0

Figura 26-Muestra del conjunto de datos Medidas consolidados. Se muestran únicamente datos agregados por motivos de privacidad de datos.

- <<Prevision unitaria meteo>>:
Estos datos consisten en previsiones de irradiación solar, convertidas a generación eléctrica fotovoltaica, considerando una potencia instalada base de generación de 100.000kW.

	CodVariable	PotInstalada	Provincia	Año	Mes	Dia	Hora	Produccion
0	1	100000	ALAVA	2018	2	1	8	666
1	1	100000	ALAVA	2018	2	1	9	4139
2	1	100000	ALAVA	2018	2	1	10	10119
3	1	100000	ALAVA	2018	2	1	11	16558
4	1	100000	ALAVA	2018	2	1	12	21007

Figura 27-Muestra del conjunto de datos Prevision unitaria meteo

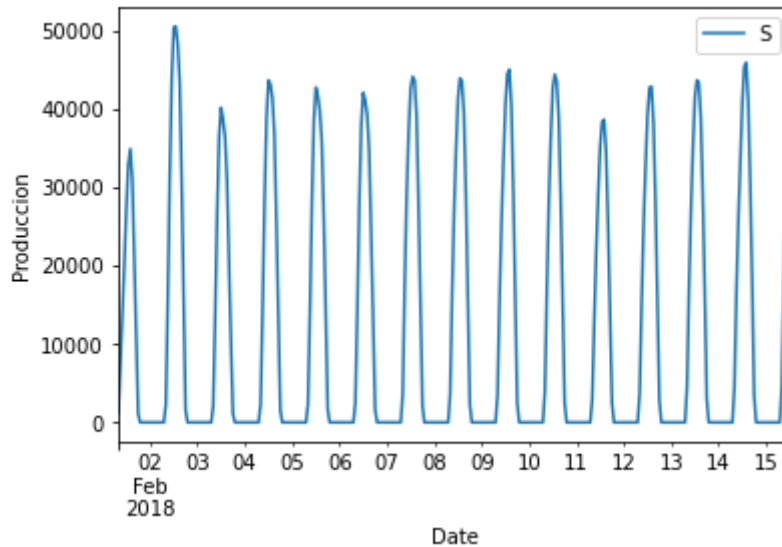


Figura 28-Produccion fotovoltaica para de una provincia. Se omiten las unidades y la provincia por motivos de proteccion de datos de clientes de Iberdrola

Este conjunto de datos está organizado en ficheros de texto con previsiones de producción fotovoltaica para una potencia de generación genérica instalada en la región. Cada fichero contiene previsiones con resolución horaria para cada una de las provincias por un periodo de dos semanas. El proveedor de datos proporciona una previsión nueva cada 12 horas.

Los ficheros provistos para la realización de este proyecto constaban de todas las actualizaciones de previsiones desde el 01/02/2018 hasta el 06/04/2021. El volumen de datos original era de 1,75GB en 2316 ficheros.

- <<Previsiones agregadas>>:

Estos datos representan previsiones enviadas por los titulares de las instalaciones a Iberdrola. Se dispone de estos datos únicamente para las unidades de generación de potencia superiores a 1MW.

	AÑO	MES	DÍA	RANGO	COMBUSTIBLE	PROVINCIA	HORA	INSTALACIONES	POTENCIA(kW)	DIARIO	...	EI3	TOTAL3	EI4	TOTAL4	EI5	TOTAL5	EI6	TOTAL6	EI7	TOTAL
0	2017	3	3	Menos de 10MW	FUEL OIL	VALENCIA	12.5	7.0	15872.0	0.000000	...	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000
1	2017	3	3	Menos de 10MW	GAS NATURAL	ALICANTE	12.5	4.0	9594.0	0.000000	...	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000
2	2017	3	3	Menos de 10MW	GAS NATURAL	CASTELLON	12.5	7.0	16302.0	0.000000	...	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.000000
3	2017	3	3	Menos de 10MW	GAS NATURAL	VALENCIA	12.5	12.0	32321.0	11.833333	...	0.0	11.833333	0.0	11.833333	0.0	11.833333	0.0	11.833333	0.0	11.833333

Figura 29- Muestra del conjunto de datos Previsiones agregadas. Se muestran datos agregados por razones de privacidad.

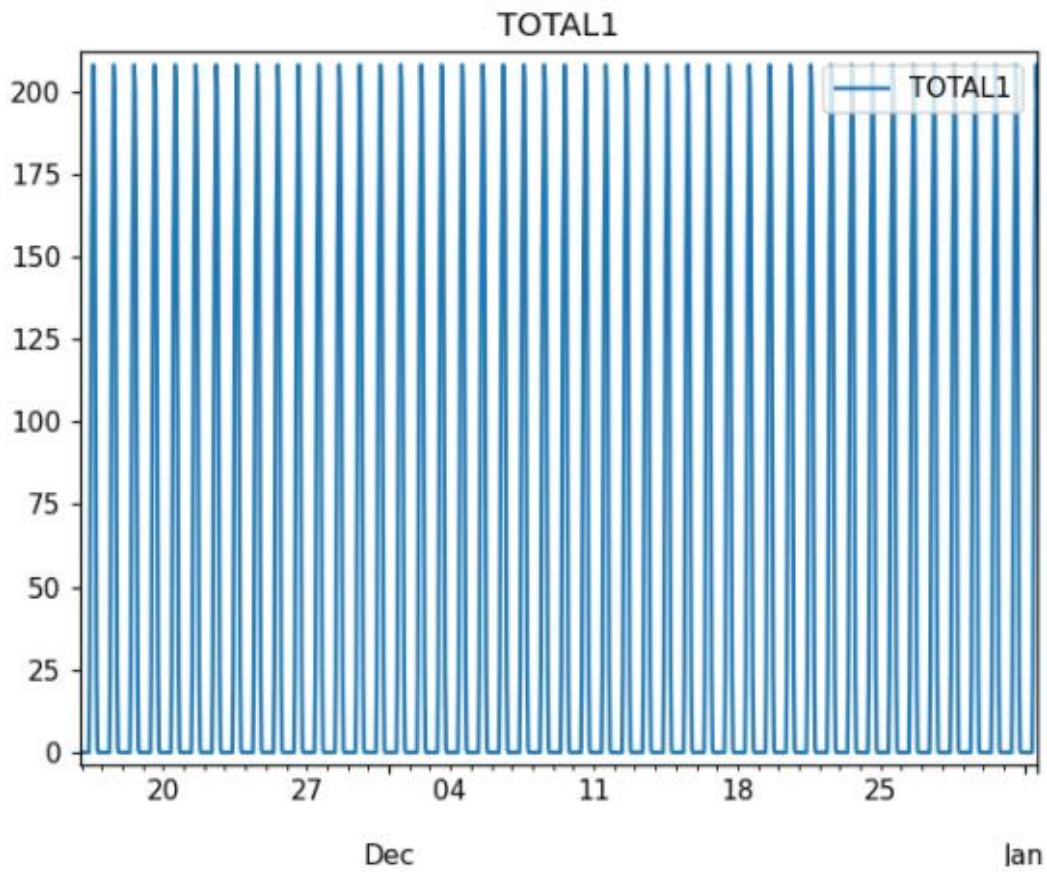


Figura 30-Ilustración de los datos contenidos en los ficheros para el dato <<TOTAL1>> y la serie de una provincia. Se omiten la provincia y las unidades por motivos de protección de datos

Casación de datos

Los datasets requeridos tienen la siguiente estructura:

- Dataset de datos fotovoltaicos:

Columnas	Descripción
Fecha	
Pruebas	True/False
Provincia	Provincia española
Potencia instalada	Float
Previsiones	Float entre 0 y 1

Energía Float

- Dataset de otras tecnologías:

Columnas	Descripción
Fecha	
Tipo de generador	Nombre de la categoría
Pruebas	True/False
Potencia	Float
Energía	Float

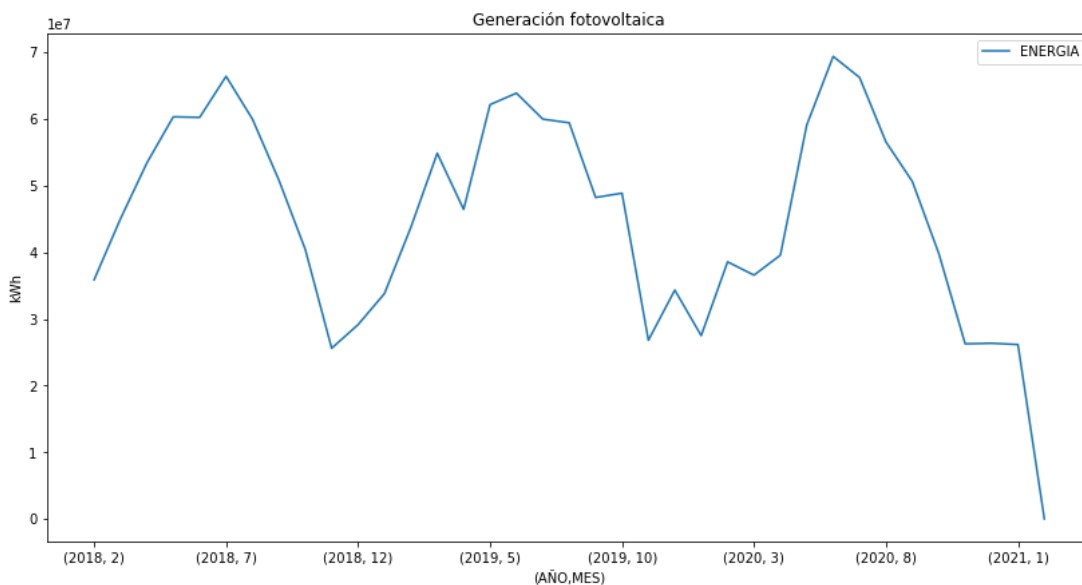


Figura 31-Generación fotovoltaica mensual de generadores sujetos a RUR

Exploración de datos y consideraciones previas al modelado

Origen de datos

Previsiones meteorológicas

Las condiciones climatológicas son indudablemente el principal condicionante de los mercados energéticos a corto plazo. Por este motivo Iberdrola dispone de diversos proveedores de previsiones meteorológicas.

Existen dos tipos de modelos meteorológicos según el tipo de extensión geográfica que cubran. Los modelos globales cubren todo el globo en un periodo futuro de entre 16 y 46 días. Estos modelos son EC¹³ o GFS¹⁴. Los modelos locales hacen previsiones de corto plazo regionales, como DWD¹⁵, AROME¹⁶, METEOSWISS o ICON-EU¹⁷

Los proveedores de datos meteorológicos proporcionan información como temperatura, precipitaciones, presión y velocidades de viento, nubosidad o radiación para localizaciones seleccionadas en formato de malla. Estos datos se interpolan y ajustan para diferentes localizaciones de interés.

Los datos obtenidos de las estaciones meteorológicas es el principal input para el cálculo de previsiones meteorológicas.

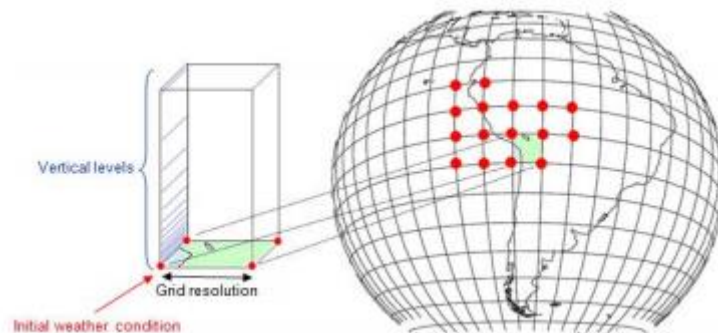


Figura 32-Illustración simplificada de un modelo de previsión meteorológica. Fuente: Refinitiv

Los datos que alimentan el modelo caracterizan el globo terráqueo o parte de este, y lo dividen en una malla regular donde cada punto tiene unas condiciones meteorológicas iniciales recabadas por estaciones meteorológicas, globos meteorológicos, aviones comerciales, embarcaciones, satélites o, cuando no existen medidas disponibles, se obtienen a partir de métodos de interpolación. La resolución de la malla varía según el modelo que se utilice que además presenta una resolución vertical (ver figura). La resolución vertical es más densa a cotas próximas a la superficie terrestre. Los modelos de previsión meteorológica calculan las previsiones para cada punto de la malla a partir de ecuaciones físicas.

En general, existen dos tipos de modelos meteorológicos:

- Operacionales (Op): Estos modelos son deterministas. Son modelos de alta resolución que proveen siempre los mismos resultados para unos mismas condiciones iniciales.
- Modelos de conjunto: Estos modelos son probabilistas con una resolución más baja que los operacionales. La característica probabilista significa que el modelo usa un conjunto de condiciones iniciales que consisten en variaciones entorno a

¹³ <https://www.ecmwf.int/en/about>

¹⁴ <https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forcast-system-gfs>

¹⁵ https://www.dwd.de/EN/Home/home_node.html

¹⁶ <https://www.umr-cnrm.fr/spip.php?article120&lang=en>

¹⁷ <https://www.ecmwf.int/en/elibrary/13229-icon-icosahedral-nonhydrostatic-modelling-framework-dwd-and-mpi-m>

las condiciones iniciales procedentes de mediciones. Las condiciones iniciales sin modificar proveen simulaciones de control. Usa resoluciones más bajas que las simulaciones operacionales pero los resultados son equivalentes. Este tipo de métodos proporciona varios escenarios y las diferencias entre distintas simulaciones da una estimación de la inestabilidad de los patrones meteorológicos actuales.

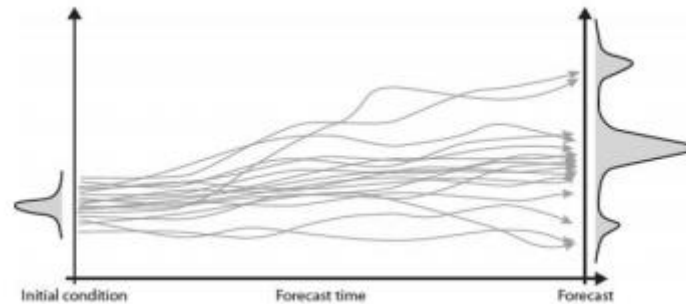


Figura 33-Representación gráfica del conjunto de simulaciones de un modelo de conjunto. Fuente: Refinitiv

Los modelos que forman las previsiones de generación fotovoltaica por potencia de referencia instalada se alimentan de datos de los proveedores de la lista siguiente:

- ECMWF: es el centro europeo de previsiones meteorológicas de alcance medio y es un proyecto comunitario europeo para proveer las mejores previsiones meteorológicas para todo el globo. Para predecir patrones de comportamiento meteorológico utilizan modelos de corrientes atmosféricas y oceánicas, modelos de perturbaciones para modelar procesos climatológicos, así como aclarar incertidumbre consecuencia de las propias herramientas de modelado.
- GFS: Es el sistema de previsiones globales del servicio americano de meteorología (NCEP – National Centers for Environmental Prediction).
- DWD: Es una institución pública alemana dependiente del Ministerio Federal de Transporte e Infraestructura Digital de Alemania. Sus predicciones se hacen con un sistema que consta de tres partes. Un modelo global que cubre toda la tierra y dos modelos locales anidados dentro del global, usando sus datos como condiciones en la frontera del dominio de definición.
- AROME: provee previsiones locales de corto plazo cubriendo Noruega, Suecia, Dinamarca, Finlandia y los países del mar Báltico. Las condiciones de contorno las proporciona ECMWF.

Performance de las previsiones meteorológicas

Existe incertidumbre en torno a la fiabilidad de las previsiones meteorológicas que procede no solo de la resolución del modelo, sino también de los procesos físicos aplicados a las condiciones iniciales y del número de observaciones. Las observaciones de las condiciones iniciales no están disponibles para cada punto de la malla. Por este motivo los proveedores de datos meteorológicos tienen que hacer interpolaciones iniciales para derivar los puntos iniciales del modelo.

El modelo de ECMWF, que es el que provee las condiciones de contorno para el cálculo de las previsiones de mayor resolución sobre el mapa español tiene una resolución de 5km.

A pesar de que los proveedores de datos meteorológicos cubren todo el globo, para los modelos de alta resolución que utilizan esos datos es necesario hacer interpolaciones para obtener las condiciones iniciales necesarias en todos los puntos de la malla de los que se quieran simular las condiciones meteorológicas. Además, se utilizan técnicas de estimación óptima como filtros de Kalman¹⁸.

Para la obtención de condiciones iniciales en puntos de interés para los que no existan medidas ni puntos de cálculo de los proveedores de datos a partir de modelos globales el proveedor de previsiones de potencia realiza una interpolación bilineal.

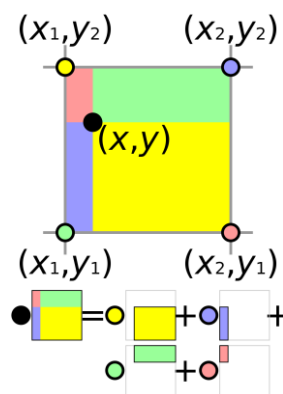


Figura 34-Illustración del método de interpolación bilineal. Fuente: https://es.wikipedia.org/wiki/Archivo:Bilinear_interpolation_visualisation.svg

Tratamiento de datos

Dask - Herramienta de tratamiento de datos

Dask se ha desarrollado con el objetivo de escalar librerías de computación científica y análisis de datos. Paquetes como NumPy, Pandas o Scikit-learn no se diseñaron para utilizarse a una escala superior a una única máquina o ejecuciones en paralelo en una máquina haciendo uso de todos los núcleos posibles de una CPU. Dask funciona con el ecosistema existente de Python para escalar a máquinas de múltiples núcleos y clústeres distribuidos.

Dask puede generar grafos de tareas donde cada nodo del grafo es una función corriente de Python y las aristas entre nodos son objetos de Python corrientes que se crean por una tarea como resultados y se usan como inputs por otra tarea. Una vez que Dask ha generado los grafos de tareas tiene que ejecutarlos en paralelo en el hardware.

¹⁸ https://es.wikipedia.org/wiki/Filtro_de_Kalman

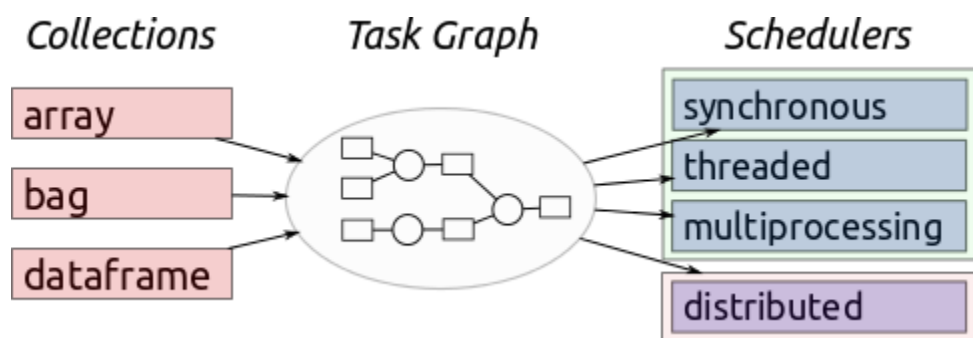


Figura 35-Ilustración del proceso de generación de grafos y ejecución en hardware. Fuente: <https://docs.dask.org/en/latest/scheduling.html>

Dask tiene dos familias de task schedulers, pero aquí solamente presentaremos los pertinentes a una sola máquina, que son los que se han utilizado en el proyecto:

- *Schedulers* de una sola máquina: Este *scheduler* provee funcionalidades básicas en un pool local de threads o de procesos.

Se va a proveer a continuación una descripción de distintas configuraciones posibles para los schedulers de una sola máquina y justificar la elección del que se usó definitivamente.

Hilos de ejecución locales (Local Threads)

Este scheduler utiliza el módulo nativo de Python `concurrent.futures`. Este módulo provee una interfaz para ejecutar funciones asíncronamente. Introduce muy poco *overhead*, de unos 50µs. Debido a que todo ocurre en el mismo proceso no tiene coste de transferencia de datos entre tareas. Como consecuencia del GIL¹⁹ (Global Interpreter Lock) de Python este scheduler solamente es capaz de proveer paralelismo en la ejecución de tareas cuando la computación está dirigida por código que no es de Python. Este es el caso cuando se trabaja con Pandas, NumPy u otros proyectos basados en C. Este scheduler es la elección por defecto de Dask.

Teniendo en cuenta que toda la preparación de datos se ha hecho con la librería `dask.dataframe`, que está construida sobre `pandas`.

Procesos de ejecución locales (Local Processes)

Este scheduler también se apoya en la librería nativa de Python `concurrent.futures`. La gran diferencia con el caso anterior es que puede ejecutar código de Python en paralelo porque todas las dependencias de las tareas de ejecución se envían a un proceso local.

Dask Distributed

Este scheduler, a pesar de su nombre, puede ejecutarse en una sola máquina. La gran ventaja de este scheduler es que provee un dashboard para el diagnóstico en tiempo real y hace una mejor gestión del data locality, siendo más eficiente que la ejecución de múltiples hilos y múltiples procesos. Esta ha sido la configuración utilizada para resolver el problema de carga de datos.

¹⁹ <https://wiki.python.org/moin/GlobalInterpreterLock>

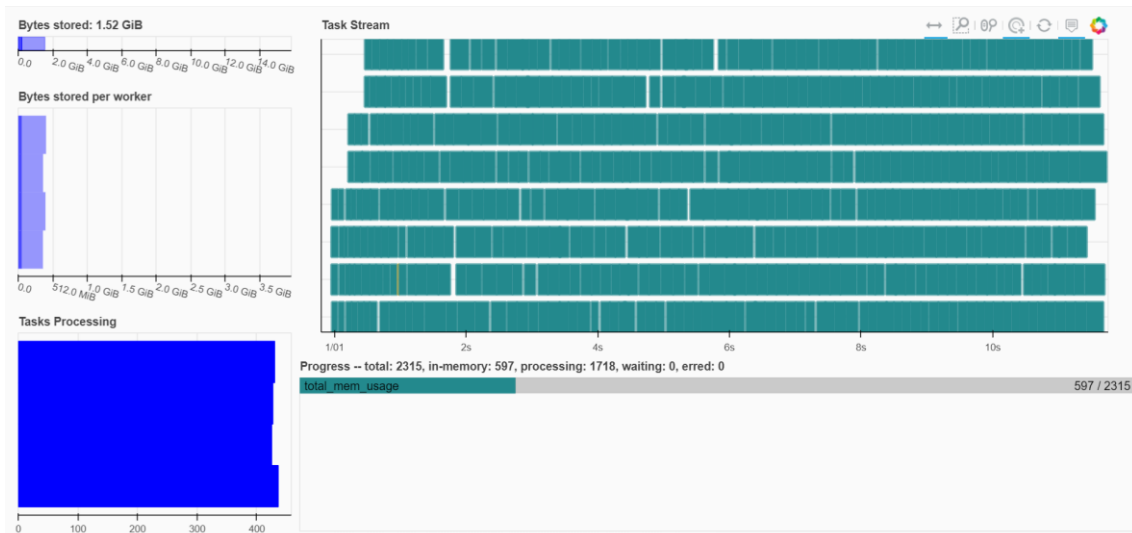


Figura 36-Illustración del dashboard de seguimiento de tareas en una máquina

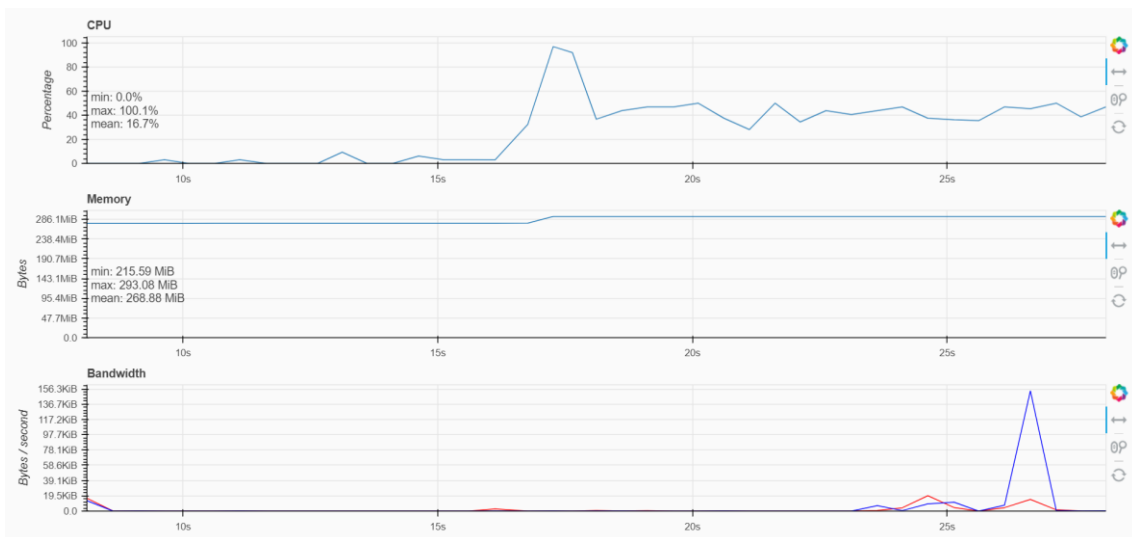


Figura 37-Illustración de los dashboards de Dask para el seguimiento de carga de memoria y CPU

Dask Dataframe

La librería `dask.dataframe` junto con `dask.distributed` han sido las librerías utilizadas para paralelizar el tratamiento de datos. Dask Dataframe es el objeto de `dask` que representa dataframes. Los dataframes de `dask` se particionan por filas, agrupándolas por valor de índice por razones de eficiencia. Cada partición de un `dask dataframe` es en realidad un dataframe de `pandas`.

Este objeto en realidad coordina múltiples dataframe de `pandas`.

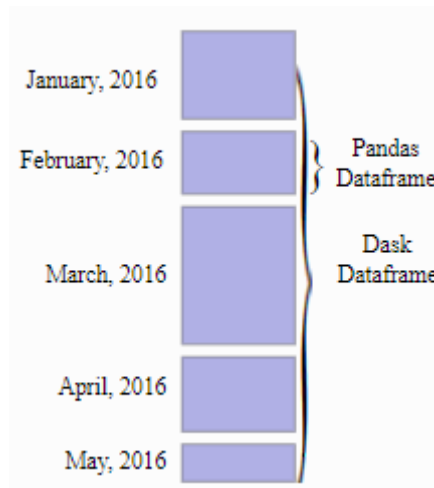


Figura 38-Illustración de la estructura de un *dask dataframe*. Fuente: <https://docs.dask.org/en/latest/dataframe.html>

Motivo de la elección de Dask frente a Spark

Dask es un software de computación distribuida. Es razonable hacerse la pregunta de por qué se ha elegido en lugar de Spark. Spark goza de una popularidad mucho mayor. El registro de consultas de *Stack Overflow* muestra que se realizan preguntas acerca de PySpark 5 veces más que de Dask en 2021. Además, Dask es un proyecto con menos historia (4 años más reciente que Spark).

Sin embargo, Dask es una herramienta que opera muy bien con Python. Está incluida en el paquete de instalación de Anaconda²⁰ y gestiona muy bien las cargas de trabajo en C/C++/Fortran/LLVM y otros códigos compilados utilizados en Python. Esto hace que potencie muy bien otras librerías como Pandas, NumPy y Scikit-learn.

Spark, sin embargo, está desarrollado en Scala y provee algún soporte para Python y R. La instalación es menos segura al depender de la JVM²¹ y la interoperabilidad con herramientas de ciencia de datos de Python de uso extendido en Iberdrola no está garantizada.

²⁰ <https://www.anaconda.com/open-source>

²¹ https://en.wikipedia.org/wiki/Java_virtual_machine

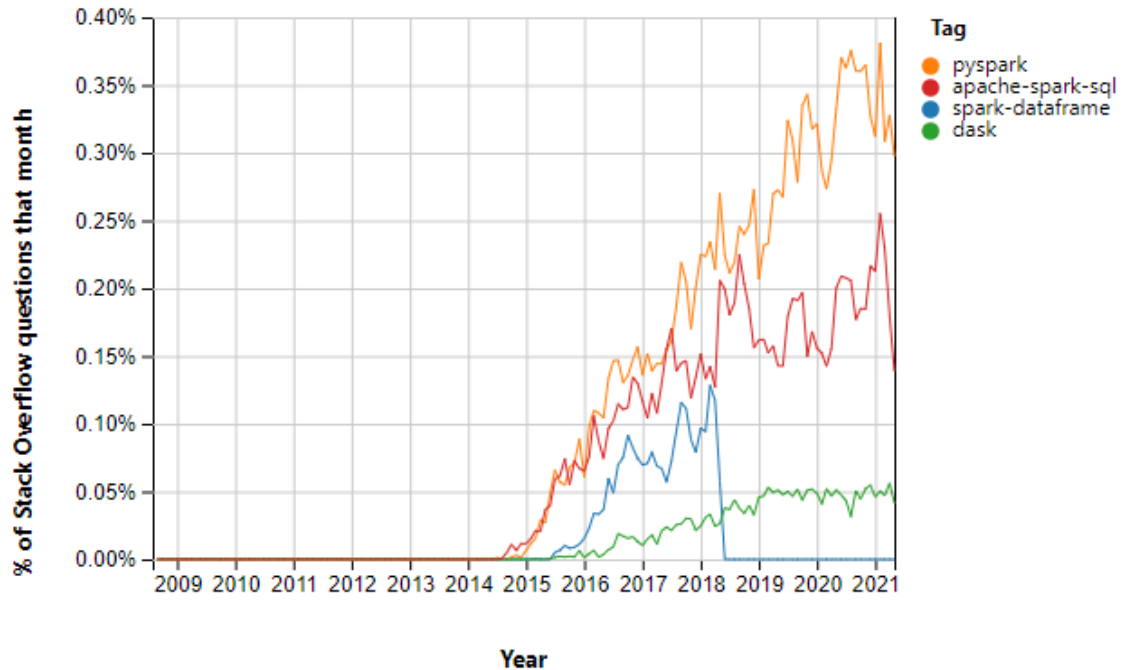


Figura 39-Comparación de popularidad entre varios frameworks de programación distribuida. Fuente: stackoverflow trends

Herramientas de modelado

DataRobot

DataRobot es una herramienta de AutoML en uso en Iberdrola. La plataforma permite entrenar modelos predictivos y hacer la puesta en producción de modelos de manera automática mediante una interfaz gráfica. Soluciona los problemas de análisis exploratorio y creación de nuevas variables explicativas. Estas características facilitan mucho la incorporación de la inteligencia artificial a los procesos de negocio, acercándola al uso diario sin requerir el conocimiento de herramientas programáticas a priori.

DataRobot organiza el proceso de modelado en el siguiente orden:

- Ingesta de datos
 - o Transformación de los datos
 - o Descubrimiento de características
- Modelado
 - o Particionamiento de datos y validación del modelo
 - o Elección de modelos para entrenar

Metodología

Al tratarse de series temporales se entrenaron modelos de regresión escogidos por la propia plataforma. El entrenamiento de los modelos se hace de manera automática en base a una medida de error seleccionada. Escogimos RMSE como medida de error, aunque a su vez, DataRobot calcula otras medidas de error para regresión. La más

significativa de todas, por motivos de interpretabilidad es MAE²², que es la medida que utilizaremos para comparar modelos.

De todos los modelos entrenados los que muestran menor error son Extreme Gradient Boosted Trees. El MAE sobre backtest es de 300.

Model Name & Description	Feature List & Sample Size	Backtest 1	All Backtests	Holdout
XG Boost eXtreme Gradient Boosted Trees Regressor with Early Stopping (learning rate =0.02) Tree-based Algorithm Preprocessing v20	Daniel_TUTO 2 years · 3 months · 18 days · 4 hours · 1 second	219.2017	314.0199	300.9207
M25 BP16 SCORING CODE ★				
XG Boost eXtreme Gradient Boosted Trees Regressor with Early Stopping (learning rate =0.02) Tree-based Algorithm Preprocessing v1	Daniel_TUTO 2 years · 3 months · 18 days · 4 hours · 1 second	219.2017	Run	300.9207
M17 BP60 SCORING CODE MONO				
XG Boost eXtreme Gradient Boosted Trees Regressor with Early Stopping (learning rate =0.02) Tree-based Algorithm Preprocessing v1	Daniel_TUTO 2 years · 3 months · 18 days · 4 hours · 1 second @ 50% Random	221.4300	319.5733	303.0968
M8 BP60 SCORING CODE MONO				

Figura 40-Tres mejores modelos por puntuación MAE

²² [Error absoluto medio - Wikipedia, la enciclopedia libre](#)

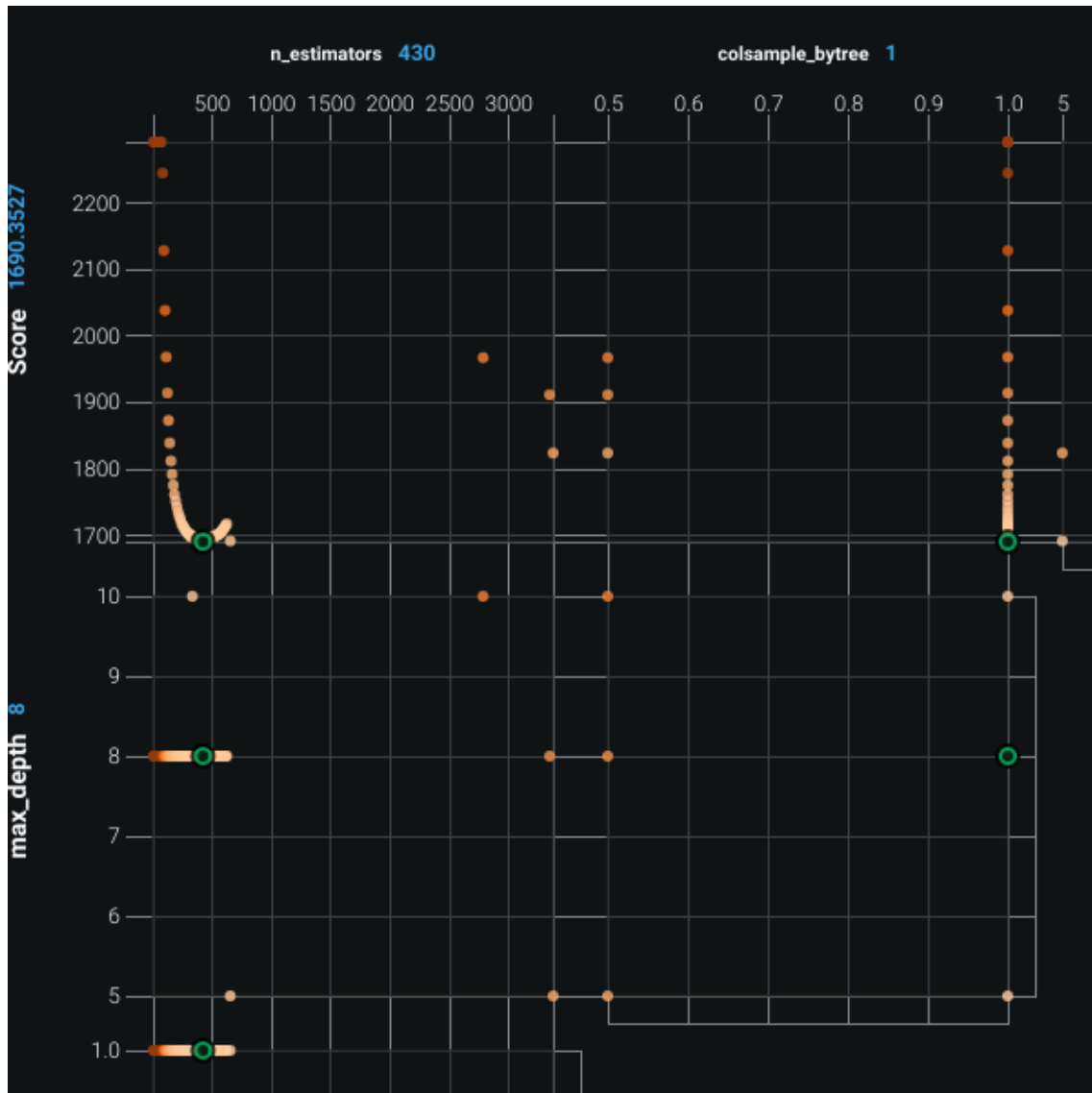


Figura 41-Sensibilidad del modelo seleccionado a los hiperparámetros

En el primer cuadrante se observa el efecto que tiene el número de árboles sobre el valor de error. El valor óptimo es 430. Un mayor número aumenta el error debido al overfitting, es decir, el modelo generaliza peor. Un modelo menos complejo no capta lo suficientemente bien la complejidad de los datos y también hace que el error sea mayor.

Estos datos se han obtenido para el hiperparámetro de profundidad máxima de árbol de 8 niveles.

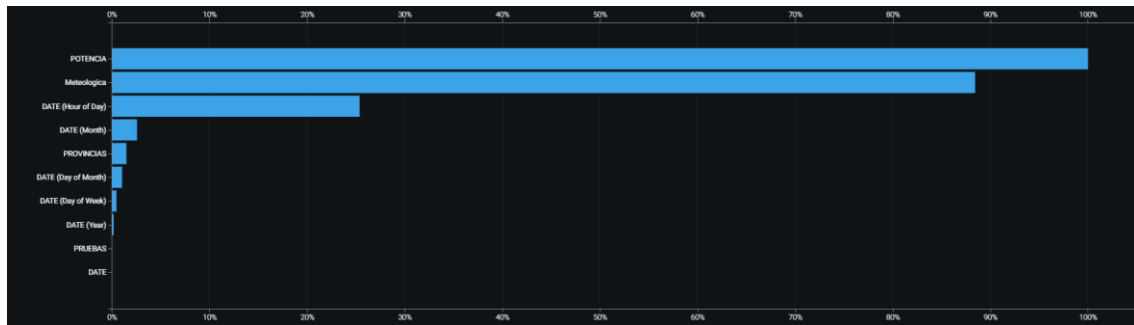


Figura 42-Importancia de las variables de entrada en la predicción del modelo

Resultados y conclusiones

- La paralelización de los procesos de lectura y tratamiento de datos reduce el tiempo de procesado de 30 minutos a 30 segundos.
- El modelo que mejores resultados proporciona es gradient boosting en el caso de series temporales fotovoltaicas para todos los algoritmos de pretratamiento utilizados.
- El software propietario de AutoML como DataRobot facilita el entrenamiento masivo de modelos. Sin embargo, al ser el código en muchas ocasiones desconocido para el usuario es complicado evaluar ciertas etapas del modelado, que quedan completamente ocultas al usuario. Suele tratarse del pretratamiento de los datos, que permanece totalmente oculto a modo de caja negra.