



UNIVERSIDAD EUROPEA DE MADRID

ESCUELA DE ARQUITECTURA, INGENIERÍA Y DISEÑO

MÁSTER UNIVERSITARIO EN BIG DATA ANALYTICS - MBI

TRABAJO FIN DE MÁSTER

**SISTEMA DE SOPORTE A LAS DECISIONES PARA
INVERTIR EN EL MERCADO DE VALORES MEDIANTE
EL USO DE PREDICTIVA ANALÍTICA**

MARC SÁNCHEZ MENAGE

CURSO 2020-2021

TÍTULO: SISTEMA DE SOPORTE A LAS DECISIONES PARA INVERTIR EN EL MERCADO DE VALORES
MEDIANTE EL USO DE PREDICTIVA ANALÍTICA

AUTOR: MARC SANCHEZ MENAGE

TITULACIÓN: MÁSTER UNIVERSITARIO EN BIG DATA ANALYTICS

DIRECTOR DEL PROYECTO: JESÚS CARRETERO

FECHA: Octubre de 2021

RESUMEN

Este proyecto está centrado en mejorar la rentabilidad de inversionistas que realizan operaciones en los mercados bursátiles. Mediante una herramienta donde el inversor tenga a mano información actualizada sobre lo que está pasando en el mercado y en sus posiciones actuales.

Aprovechando técnicas de aprendizaje automático se trata de aportar un mayor acierto en las operaciones para aumentar la rentabilidad anual. La estrategia de inversión utilizada está basada en el largo plazo concretamente en la filosofía Value Investing, que se basa en realizar un análisis de los ratios financieros de las empresas o fundamentales de las compañías. En contraposición al Trading que se basa en el análisis técnico del precio y el volumen donde las operaciones se hacen en cortos periodos de tiempo (minutos, horas o días).

El desarrollo culminará con una herramienta visual con un cuadro de mandos donde se mostrará las posiciones actuales del portfolio, rendimiento de la cartera, sugerencias de inversión basadas en predicciones por Machine Learning y un indicador de estado predictivo de compra, venta o mantener según el estado actual del mercado.

Palabras clave: Sistema de soporte a las decisiones, Value Investing, Análisis Fundamental, Kibana, Machine Learning, Bolsa

ABSTRACT

DDS decision support systems have taken on a strong importance in recent years because are tools that help organizations find market opportunities. In an increasingly globalized economy, decisions are more complex and requires tools capable of managing and analyze a greater amount of information.

This project aims to help retail investors through a Decision Support System to manage risk and reduce the volatility of their portfolios in the stock market. It is especially focused on those investors whose strategies are based on the Value Investing investment philosophy, which seeks generate positive long-term returns by buying quality assets when their price is below their intrinsic or real value. This philosophy was originated in 1928 by Benjamin Graham in his book "Security Analysis" and later popularized by his disciple Warren Buffett.

Applying this philosophy with advanced machine learning techniques. Patterns will be found to determine the real stock value using the company financial ratios and then it will be determined if the stock has growth potential, determining its real price or intrinsic value. If the system determines that the intrinsic value is above the price of the share on the stock market, it will establish that it is a cheap share and that it still has potential growth, therefore it will recommend buying more shares. Otherwise, if it establish that the intrinsic value is below the listed price, it will recommend sell until the share price falls again.

During the development of the system, three key points have been worked on. First, obtaining the financial data. It is very important to obtain complete and up-to-date data as predictions are based on this information. On a quarterly basis, companies publish their financial results, so it is very important have a system aware to the latest news since is when market opportunities are created. To perform this, an automatic data scraping has been carried out on a free website that has a Refinitiv (Reuters) database, updated daily with the financial data of most of the companies listed on the stock market worldwide.

Second, determine intrinsic value using machine learning techniques using companies financial ratios. A study has been carried out to find which ones have more predictability power and discard those variables that do not help on the prediction. In addition, a small investigation has been carried out on the most used algorithms and different strategies to improve the portfolios profitability, which have later been put into practice and a comparison has been made to determine which of the algorithms and strategies has greater predictability.

Finally, the model obtained plus the updated data has been integrated into a visual panel with Kibana, where the investor can see his portfolio shares and get buy or sell alerts. When there is a change on the shares price or there is a variation in the financial ratios that produce a change in the intrinsic value of the share.

Key words: Decision support system, Value Investing, Machine Learning, Fundamental Analysis, AI.

AGRADECIMIENTOS

En agradecimiento a mi primo Damià por compartir sus conocimientos sobre el mundo de las finanzas y las inversiones.

A mi familia por su apoyo que me ha acompañado durante todo este proceso.

A los profesores del máster y en especial a Jesús Carretero, tutor del proyecto por guiarme y aconsejarme durante el desarrollo del proyecto.

Índice

RESUMEN	4
ABSTRACT	5
Capítulo 1. INTRODUCCIÓN.....	12
1.1. Planteamiento del problema	12
1.2. Objetivos del proyecto	12
1.3. Estructura del proyecto	13
Capítulo 2. ESTADO DEL ARTE	14
2.1. La filosofía Value Investing	14
2.2. Margen de seguridad en el Value Investing	15
2.3. Diferencias entre el análisis fundamental y técnico	16
2.4. Enfoques del análisis fundamental.....	16
2.5. Métodos de valoración	17
2.6. La formula de Benjamin Graham.....	18
2.7. Calculo del valor intrínseco	19
2.8. Algoritmos de aprendizaje.....	22
2.9. Random Forest (RF)	22
2.10. Adaptative Neural Fuzzy Interface System (ANFIS).....	25
2.11. Support Vector machine (SVM).....	28
2.12. Feedforward neural network (FNN)	32
2.13. Value Investing aplicando técnicas de aprendizaje automático	35
Capítulo 3. DESARROLLO DEL SISTEMA.....	37
3.1. Introducción.....	37
3.2. Proceso de extracción de los datos	38
3.2.1. Web Scraping	39
3.2.2. Desarrollo del script de scraping.....	42
3.2.3. Creación del dataset.....	43
3.2.4. Obtener datos del bróker	44
3.2.5. Obtener posiciones del bróker.....	44
3.2.6. Obtener rendimiento de la cartera	46
3.3. La base de datos	48
3.3.1. Configuración del sistema	49
3.3.2. Almacenamiento con Elasticsearch	52
3.4. Algoritmo de aprendizaje automático.....	55
3.4.1. Limpieza de datos.....	56

3.4.2.	Correlación y estudio de las variables.....	56
3.4.3.	Selección de los algoritmos.....	58
3.4.4.	Aplicación del algoritmo al script.....	59
3.5.	Visualización de los datos con Kibana.....	61
3.5.1.	Diseño del dashboard.....	61
3.5.2.	Configuración del dashboard.....	63
3.5.3.	Visualización de los datos.....	64
Capítulo 4.	CASO DE NEGOCIO.....	67
4.1.	Introducción.....	67
4.2.	Estudio de mercado.....	67
4.3.	Resumen de mercado.....	70
4.4.	Plan de negocio.....	71
Capítulo 5.	PRESUPUESTO.....	73
Capítulo 6.	PLANIFICACIÓN.....	74
Capítulo 7.	CONCLUSIONES Y FUTURAS LÍNEAS DE TRABAJO.....	75
ANEXOS	76
BIBLIOGRAFÍA	77

Índice de Figuras

Ilustración 1 Margen de seguridad	15
Ilustración 2 Arquitectura de un algoritmo Random Forest.....	22
Ilustración 3 Modelo difuso Takagi-Sugeno.....	25
Ilustración 4 Arquitectura completa del modelo ANFIS	26
Ilustración 5 Ejemplo de separación de clases mediante el vector soporte	28
Ilustración 6 Representación de la Ecuación del Hiperplano y Ecuación de Línea.....	29
Ilustración 7 Muestra del hiperplano óptimo.....	30
Ilustración 8 Representación gráfica de los límites máximos de los hiperplanos	30
Ilustración 9 Representación de los márgenes D1 y D2	31
Ilustración 10 Arquitectura de una Red Neuronal de Retroalimentación	32
Ilustración 11 Modelo matemático de un perceptrón	33
Ilustración 12 Ratios financieros en app.tikr.com	38
Ilustración 13 Arquitectura del script dataset_extractor	43
Ilustración 14 Arquitectura del script portfolio	44
Ilustración 15 Recomendaciones según el margen de seguridad	45
Ilustración 16 Rendimiento cartera vs SP500	46
Ilustración 17 Arquitectura del script performance	46
Ilustración 18 Componentes de Elastic Stack	48
Ilustración 19 Configuración AWS: Tipo de instancia	49
Ilustración 20 Configuración AWS: Tipo de almacenamiento	49
Ilustración 21 Configuración AWS: Nodos máster.....	50
Ilustración 22 Configuración AWS: Control de acceso.....	50
Ilustración 23 Configuración AWS: Datos de acceso al nodo	51
Ilustración 24 Arquitectura del script updater	52
Ilustración 25 Esquema del índice predictions	54

Ilustración 26 Dataset con los datos financieros	55
Ilustración 27 Limpieza de datos	56
Ilustración 28 Matriz de correlación Pearson	56
Ilustración 29 Las diez variables con mayor correlación	57
Ilustración 30 Arquitectura del script predictions	59
Ilustración 31 Kibana: Tipos de visualizaciones disponibles	61
Ilustración 32 Distribución del dashboard	61
Ilustración 33 Kibana: Index Patterns configurados	63
Ilustración 34 Kibana: Campos de los índices	63
Ilustración 35 Kibana: Portfolio por países	64
Ilustración 36 Kibana: Resumen del portfolio.....	64
Ilustración 37 Kibana: Portfolio vs SP500	65
Ilustración 38 Kibana: Sugerencias de inversión.....	65
Ilustración 39 Kibana: Posiciones abiertas.....	66
Ilustración 40 Planificación del proyecto	74

Índice de Tablas

Tabla 1 Diferencias entre análisis fundamental y análisis técnico	16
Tabla 2 Estados financieros de Apple del año 2020	20
Tabla 3 Proyección del Free Cash Flow a cuatro años	20
Tabla 4 Funciones de activación de una red neuronal	34
Tabla 5 API REST: Buscar compañía y obtener datos básicos.....	39
Tabla 6 API REST: Obtener RIC: Reuters Instrument code	39
Tabla 7 API REST: Obtener datos financieros	40
Tabla 8 API REST: Obtener lista de accionistas	40
Tabla 9 API REST: Obtener precio histórico	41
Tabla 10 Arquitectura del script de scraping	42
Tabla 11 Estructura del fichero portfolio.json	45
Tabla 12 Comandos del script updater	52
Tabla 13 Índices de Elasticsearch.....	53
Tabla 14 Estadísticas de los modelos entrenados	58
Tabla 15 Presupuesto del proyecto	73

Capítulo 1. INTRODUCCIÓN

1.1. Planteamiento del problema

En los últimos años se han realizado numerosos estudios [1] [2] [3] sobre la predicción de valores mediante el uso del aprendizaje automático. La mayoría de ellos, están enfocados en la predicción a corto plazo mediante el uso indicadores técnicos y del precio histórico de una acción. Aunque la mayor parte de estudios se centran en análisis técnico, existen otras técnicas como el análisis fundamental y el análisis de sentimiento que no han sido tan explotadas.

El análisis técnico destaca principalmente porque sus datos son de fácil acceso, ya que los indicadores técnicos están basados únicamente en los precios y en el volumen histórico. Podemos encontrar numerosas bases de datos en la red donde podemos extraer toda esta información. En cambio el análisis fundamental se basa en los estados financieros de una compañía que se publican de forma trimestral y oficialmente en la página web de la Comisión de Bolsa y Valores de cada país. El mayor reto que plantea, es que la información no se encuentra homogeneizada ni armonizada lo que lo hace sumamente complicado poder desarrollar un proyecto de Inteligencia Artificial.

Gracias al auge de plataformas de brokerage online [4] y en parte por el boom de las inversiones en criptomonedas este sector ha experimentado un fuerte cambio en los últimos años, lo que ha empujado a empresas del sector financiero a recoger toda la información de los estados financieros de las empresas en plataformas online. Podemos obtener información homogénea y actualizada prácticamente al instante de cualquier empresa del mundo y descargar el histórico de cada trimestre. Este punto de inflexión ha propiciado el poder desarrollar esta tesis de máster sobre un sistema de soporte a las decisiones basada en inteligencia artificial.

1.2. Objetivos del proyecto

El objetivo de este proyecto es desarrollar un sistema de ayuda a la toma de decisiones para pequeños inversores, donde puedan encontrar nuevas oportunidades de mercado y realizar un seguimiento de sus acciones. Mediante un sistema de alertas el inversor será capaz de identificar los cambios que se producen en el mercado y a través de una IA le recomendará que operaciones realizar con cada sus acciones, informando si es buen momento de comprar o vender las acciones.

El desarrollo se realizará sobre un sistema de código abierto con la herramienta Kibana [5] que tiene la capacidad de crear visualizaciones y búsqueda de datos, proporcionando interfaces visuales muy intuitivas y sencillas que ayudan al usuario entender rápidamente la información.

Los datos se obtendrán de app.tikr [6], un proveedor de datos del mercado financiero mediante un script de Python que se ejecutará de forma diaria y actualizará la base de datos de datos de Elasticsearch.

Además se realizará un estudio con distintos algoritmos de aprendizaje automático y distintas estrategias para descubrir que ofrece una mayor capacidad de acierto para más tarde implementarlo en el sistema de soporte y obtener las recomendaciones.

1.3. Estructura del proyecto

Este proyecto está dividido en tres grandes bloques, el primero es una **introducción al mundo financiero**, concretamente al Value Investing que se utiliza como punto de partida teórico en el desarrollo de los algoritmos de aprendizaje automático y una investigación de los algoritmos más utilizados en este tipo de inversión

El segundo bloque se centra en el **desarrollo de un sistema de toma de decisiones** para invertir en el mercado de valores enfocado a inversores que quieren mejorar sus retornos gracias a las ventajas que ofrece de la inteligencia artificial, este bloque está formado por sub-fases.

La primera fase se ha centrado en la **obtención de los datos** financieros de las compañías. Para ello se ha utilizado una página web gratuita que ofrece los datos de los mercados bursátiles más importantes del mundo. Esta fase es una de las más importantes ya que se requieren datos reales y fidedignos para conseguir un sistema fiable.

Una vez obtenidos los datos se ha utilizado una base de datos para **almacenar la información**, se ha elegido para ello Elasticsearch principalmente por la facilidad de poder guardar y leer datos desde Python con su librería oficial. Además de las funcionalidades de visualización que posteriormente facilitan el trabajo de creación de un dashboard.

Con los datos en Elasticsearch se ha creado un script para **generar un dataset** con los ratios que se han considerado más importantes y que podrían aportar mayor poder predictivo. Centrándolo en un periodo comprendido entre 2015 y 2021 para aportar mayor información estadística al sistema.

A continuación se ha utilizado el dataset para realizar un **análisis en un notebook**, donde se ha realizado una limpieza de los datos para su posterior estudio.

Tras la **limpieza de datos**, se han aplicado distintos **algoritmos de aprendizaje automático** y se ha evaluado su desempeño. Con el que se ha obtenido mayor eficacia se ha exportado el modelo para ser aplicado más tarde en el script de Python.

Para mantener al día el sistema se ha desarrollado un script llamado updater.py que descarga los datos financieros de forma mensual y **actualiza el portfolio diariamente y las predicciones** mensualmente. Los datos del portfolio los obtiene directamente del bróker DEGIRO.

Finalmente con los datos actualizados en la base de datos Elasticsearch, se ha utilizado Kibana, que forma parte del framework de Elastic para poder realizar la **visualización de datos** que nos ayudará a tomar las decisiones.

El tercer y último bloque está centrado en el **plan de negocio** del proyecto donde se hace un estudio de los competidores, público objetivo, precio y coste del servicio, presupuesto y planificación.

Capítulo 2. ESTADO DEL ARTE

2.1. La filosofía Value Investing

El Value Investing [7] es una filosofía de inversión que nació en Estados Unidos de la mano de **Benjamin Graham**. En el año 1928 publicó el libro ***Security Analysis***, donde planteaba esta filosofía y metodología de inversión, que varias décadas después se popularizó por su discípulo **Warren Buffet**, gran conocido en el mundo de las inversiones por sus grandes retornos financieros aplicando esta filosofía.

La inversión en valor o Value Investing consiste en comprar acciones de calidad a un precio infravalorado, es decir se trata de comprar valores de buenas compañías que se encuentran a un precio inferior de mercado, con la premisa de que los mercados financieros serán **más eficientes a largo plazo**.

La bolsa a corto plazo puede llegar a ser irracional y comportarse de una forma inesperada, generando cotizaciones muy bajas o excesivamente altas, pero con el tiempo el mercado se estabiliza y el activo toma el precio que le corresponde.

Por lo tanto, si compramos acciones en el momento en que están **infravaloradas** podemos obtener grandes retornos cuando el precio de la acción se sitúe en su valor real. Los inversores value investors rentabilizan su dinero seleccionando compañías sólidas, con potencial de crecimiento y un buen equipo gestor que garantice unos retornos recurrentes a largo plazo.

Para poder establecer un precio teórico o **valor intrínseco** de un activo se utilizan distintos métodos de análisis financieros, el más conocido es el **análisis fundamental** que trata de establecer el precio mediante el estudio de los estados financieros, planes de expansión, sector y entorno económico de una compañía.

2.2. Margen de seguridad en el Value Investing

El margen de seguridad [8] es una de las herramientas más utilizadas por los inversores value, permite **reducir el riesgo de las inversiones** y obtener mayores retornos. Este termino fue acuñado por Benjamin Graham en su libro “El inversor inteligente”.

Se define como la diferente entre el precio y el valor intrínseco, si el valor intrínseco no varia y el precio de un activo es menor que el primero, el margen de seguridad será mayor y por lo tanto el riesgo que asumiremos será más bajo.



Ilustración 1 Margen de seguridad

Para calcular el margen de seguridad se aplica la siguiente formula:

$$\text{Margen de seguridad} = \frac{\text{Precio de mercado}}{\text{Valor intrínseco}} \cdot 100$$

Es decir si una empresa cotiza a 30\$ por acción y después de estudiar las cuentas anuales determinamos que su valor intrínseco es de 50\$ por acción el margen de seguridad que obtendríamos es de un 60% o lo que es lo mismo un descuento del 60% por su futura revaloración.

Por supuesto un alto margen de seguridad no garantiza que perdamos dinero, pero disminuye el riesgo ante por ejemplo cambios en el sector, crisis económicas o eventos inesperados de la compañía. Además de la reducción del riesgo nos puede aportar una mayor rentabilidad en futuro, ya que si elegimos aquellas empresas con mayor margen de seguridad serán las que mas potencial de revaloración tendrán y por consiguiente a largo plazo obtendremos más retornos que si eligiéramos las de menor margen de seguridad.

2.3. Diferencias entre el análisis fundamental y técnico

El análisis fundamental y técnico [9][10] son los dos tipos análisis bursátiles que tratan de predecir el comportamiento del mercado financiero. El primero trata de predecir el precio mediante sus estados financieros o noticias y el segundo mediante datos estadísticos del precio.

En la siguiente tabla se recogen las principales diferencias entre el análisis fundamental y técnico.

	ANÁLISIS FUNDAMENTAL	ANÁLISIS TÉCNICO
Filosofía	Se calcula el valor de un activo y se compra si está infravalorado, para ganar hay que acertar en el valor del activo y esperar a que converja.	El mercado aporta la mejor información sobre la evolución del precio, para ganar hay que salir antes de que cambie la tendencia.
Objetivo	Determinar si el activo está sobrevalorado o infravalorado	Encontrar el mejor momento de entrada y salida del mercado
Otras denominaciones	Value investing	Trading
Horizonte de tiempo	Largo plazo (meses, años)	Corto plazo (minutos, horas días)
Perfil del inversor	Crítico, paciente, mantiene la calma	Rápido, metódico y con instinto de jugador

Tabla 1 Diferencias entre análisis fundamental y análisis técnico

2.4. Enfoques del análisis fundamental

Las variables que afectan valor de una acción pueden ser **microeconómicas** o **macroeconómicas**, las primeras son propiamente de cada empresa y tienen que ver con el desempeño que realizan las corporaciones, los balances, los estados financieros y los comunicados. Las macroeconómicas se centran en la tendencia general de los mercados, el estado de la economía general y del país donde operan, el nivel de desempleo y el PIB.

Existen dos tipos de enfoque de análisis fundamental:

- El **método Top-Down**: Se realiza un análisis de arriba abajo, es decir se empieza por lo general (macroeconomía) hasta lo particular o local (microeconomía). Se estudia la situación económica mundial, que países existe mayor atractivo, que sectores son los más importantes de ese país y dentro de esos sectores se analizan aquellas empresas que tengan mayor potencial.
- El **método Bottom-Up**: Se realiza un análisis que va de lo particular (microeconomía) a lo general (macroeconomía). Primero se eligen las empresas que puedan crecer más, se realiza un análisis del sector, se estudia la situación económica del país y finalmente se analiza la situación económica global.

2.5. Métodos de valoración

Los métodos de valoración [11] son independientes al enfoque que elige el analista, para valorar una empresa existen numerosos métodos de valoración cuantitativa que permiten calcular el precio de una compañía, pero para ello es muy importante conocer la diferencia entre **precio** y **valor**. En la siguiente cita Warren Buffet ilustra el significado entre precio y valor:

"Precio es lo que pagas, valor es lo que recibes" – Warren Buffet

Con ello manifiesta que el precio se puede cuantificar de forma objetiva ya que se trata del **dinero** que hay que pagar para adquirir un bien o producto y depende de la oferta y la demanda. En cambio, valor es la **utilidad** que se recibe a cambio, que puede ser la satisfacción que obtiene el comprador o de unos beneficios económicos que pueda obtener.

Los principales métodos de valoración de empresas son:

- **Métodos basados en el balance:** Es un método estático de valoración, que calcula el valor presente de una empresa mediante el patrimonio, es un método que no tiene en cuenta la evolución futura de crecimiento de una empresa, el valor temporal del dinero, su situación actual o sus recursos. Es por ello un método poco fiable respecto a otros. Algunos métodos de este tipo son: valor contable, valor ajustado, valor de liquidación, valor en libros valor sustancial.
- **Métodos basados en ingresos declarados:** Es un método de valoración de empresas que determina su valor en función de los resultados de una compañía. Utilizando las dimensiones de sus ganancias, ventas y otros múltiplos como por ejemplo el PER, EBIT, EBITDA, múltiplos de venta y dividendos.
- **Métodos basados en el fondo de comercio:** Se valora una empresa según sus intangibles, es decir se obtiene cuantificando todos los activos inmateriales que poseen como por ejemplo patentes, marcas, cartera de clientes, liderazgo en la industria, ventajas competitivas. Entre los distintos métodos encontramos: el clásico, unión de expertos, renta abreviada, método indirecto.
- **Métodos basados en el descuento de flujos de caja:** Es el método más utilizado por los inversores. Se valora una empresa en función de la capacidad para generar dinero en el futuro, contempla el valor temporal del dinero y considera las inversiones requeridas para la generación de recursos, permite determinar el **valor intrínseco**. Los flujos se pueden clasificar en: Flujos de caja financiero, flujos de caja por acción, dividendos, flujo de caja del capital.

2.6. La formula de Benjamin Graham

Benjamin Graham ideó una formula [12] para calcular el valor intrínseco de una compañía, aunque esta tiene ciertas limitaciones permite hacer una aproximación. La formula es la siguiente:

$$V = \frac{EPS * 4.4 * (8.5 + 2g)}{y}$$

EPS: Earnings per Share (ganancias por acción)

g: Estimación del crecimiento de ganancias

y: AAA Corporate Bond Yield % actual

8.5: Graham's P/E ratio para una empresa sin crecimiento

4.4: AAA Corporate Bond Yield en 1962

Como se puede observar la formula es muy simple por lo que tiene ciertos inconvenientes, en primer lugar no utiliza el endeudamiento, siendo un factor muy importante a la hora de valorar el riesgo de una empresa. Una empresa que no tenga deuda se debe valorar más positivamente que una que tenga cierto apalancamiento, si ambas tienen la misma previsión de crecer.

Otros factores como el posicionamiento del mercado, análisis competitivo, sector no se tienen en cuenta en esta formula por lo que realizar un análisis fundamental con sólo estos datos sería incompleto y es necesario realizar un análisis cualitativo que debería realizar todo inversor antes de invertir.

Para ejemplificar el procedimiento se va a analizar The Coca-Cola Company. Se obtienen los ratios de la compañía de la última publicación anual (año 2020), que se utilizarán para el análisis:

	Año 2020
EPS: Earning per Share	1.78
<i>g</i> : tasa de crecimiento de las ganancias media	2.9
<i>y</i> : índice Corporate Bond Yield	2.55

$$V = \frac{1.78 * 4.4(8.5 + 2 * 2.9)}{2.55} = \$43.92$$

El valor intrínseco de la compañía es \$43.92 si el precio de la acción actual es de \$53.89 estaría sobrevalorada y no sería un buen momento para comprar acciones

2.7. Cálculo del valor intrínseco

Para calcular el valor intrínseco [13] de una compañía, el método de descuento de caja [14] es uno de los más utilizados para valorar empresas. A continuación, se detalla el procedimiento para obtenerlo. Primero debemos calcular el descuento de flujos de caja o también conocido por Free Cash Flow (FCF) de 5 años anteriores para ver la tendencia de crecimiento, para ello se aplica la siguiente fórmula:

$$FCF = CFO - CAPEX$$

CFO: Flujo de caja operativo - Dinero efectivo que genera la empresa a través de sus operaciones y el ejercicio de su actividad

CAPEX: Inversiones de capital - Compra de inmuebles o la adquisición de vehículos

Ambos valores son fáciles de obtener ya que se publican en los informes trimestrales de las compañías. Aunque a veces hay que calcular el flujo de caja operativo que se obtiene de la siguiente forma:

$$CFO = EBIT + Amortizaciones - Impuestos$$

EBIT: Ganancias antes de Intereses e Impuestos

Amortizaciones: Costes de una deuda por medio de pagos periódicos

Impuestos: Tributos a pagar a las administraciones públicas

Una vez obtenido el free cash flow de los 5 años anteriores comprobamos su tasa de crecimiento anual y obtenemos la media porcentual del crecimiento (FWD), que más tarde utilizaremos. A continuación se muestra la fórmula del valor intrínseco, primero se suma de las proyecciones a n años del flujo de caja, teniendo en cuenta el crecimiento anual que ha experimentado en años anteriores dividido por la tasa de descuento. Finalmente, se suma el valor terminal que es la estimación del crecimiento de flujos de efectivo esperados más allá del período de proyección.

$$IV = \overbrace{\frac{FCF_1}{(1+r)^1} + \dots + \frac{FCF_n}{(1+r)^n}}^{\text{Proyección}} + \overbrace{\frac{FCF_n}{g}}^{\text{Valor Terminal}}$$

IV: Intrinsic Value (Valor intrínseco)

FCF_n: Flujo de caja de año n

r: Tasa de descuento

n: Número de años que vamos a analizar

g : Tasa de crecimiento terminal esperada (siempre es inferior a r)

Una vez obtenido el valor intrínseco total, sólo hay que dividirlo por el número de acciones en circulación y obtendremos el precio de la acción.

Para ejemplificar el procedimiento se va a analizar la compañía Apple. Se obtienen los ratios de la compañía de la última publicación anual (año 2020), que se utilizarán durante el análisis:

	Año 2020
FWD (Free Cash Flow Per Share Growth Rate) 5 years avg.	4.68%
CFO (Cash from Operations)	\$80,674.0
CAPEX (Capital Expenditure)	\$7,309.0
Total Common Shares Outstanding	16,976.8

Tabla 2 Estados financieros de Apple del año 2020

Se calcula el free cash flow a año 2020 y se proyecta para los próximos cuatro años partiendo de un crecimiento del 4.68% anual y asumiendo que la tasa de crecimiento terminal será del 3.8%.

$$FCF = 80,674 - 7,309 = 73,365$$

	2020	2021	2022	2023	2024
FCF	\$73,365.00	\$76,798.48	\$80,392.65	\$84,155.03	\$88,093.48
Tasa de descuento	1.03	1.06	1.09	1.13	1.16
Valor presente	\$71,228.16	\$72,389.93	\$73,570.66	\$74,770.65	\$75,990.21

Tabla 3 Proyección del Free Cash Flow a cuatro años

Se calcula el Terminal Value:

$$TV = \frac{FCF_n}{g} = \frac{88095.5}{3.8} = \$2,318,249.53$$

Se obtiene el valor intrínseco de la compañía:

$$IV = 71,228.16 + 72,389.93 + 73,579.66 + 74,770.65 + 75,990.21 + 2,318,249.53$$

$$IV = \$2,283,298.81$$

Para obtener el precio por acción se divide por el número de acciones en circulación:

$$IV \text{ per share} = \frac{IV}{Shares} = \frac{\$2,283,298.81}{16,976.8} = \$134.50$$

Es decir el valor intrínseco de la empresa Apple es de \$134.50, actualmente cotiza a \$146.06 por lo que estaría sobrevalorada y no sería un buen momento para comprar acciones.

Como se puede observar el valor intrínseco tiene un **punto subjetivo**, donde de las hipótesis que se toman pueden hacer variar su resultado y por lo tanto no se puede tomar como un valor exacto sino como más bien como una aproximación basada en la experiencia del análisis de empresas.

Estos cálculos deben de ir acompañados de un **análisis cualitativo** del modelo del negocio, aunque en este proyecto no se va a tratar ya que el objetivo de este, es mejorar los **análisis cuantitativos** mediante el uso de aprendizaje automático. El público objetivo son pequeños inversores que tratan de mejorar sus retornos por lo que se tratará de proporcionarles una herramienta de soporte al trabajo que ya realizan.

Además, ayudará a filtrar y enfocar los esfuerzos en aquellas empresas que merezcan más la pena realizar un análisis, poniendo el foco en aquellas acciones q se puedan obtener un mayor retorno y se encuentren infravaloradas y por consiguiente tengan un mayor margen de seguridad.

2.8. Algoritmos de aprendizaje

A continuación se describe los algoritmos de aprendizaje más utilizados para el análisis fundamental de empresas.

2.9. Random Forest (RF)

Random Forest [15] [16] [17] es un algoritmo de aprendizaje de clasificación y regresión que utiliza árboles de decisión en la fase de entrenamiento.

Los árboles de decisión son generados con el método de bagging donde los algoritmos simples son usados en paralelo. El principal objetivo de los métodos en paralelo es el de aprovecharse de la independencia que hay entre los algoritmos simples, ya que el error se puede reducir al promediar las salidas de los modelos simples para llegar a una decisión más estable y precisa.

Su principal ventaja respecto a otros algoritmos es que tiene un mayor rendimiento de generalización, que consiste en compensar los errores de las predicciones de los árboles de decisión para asegurar que los árboles son distintos y se entrenan cada uno con una muestra aleatoria de datos de entrenamiento.

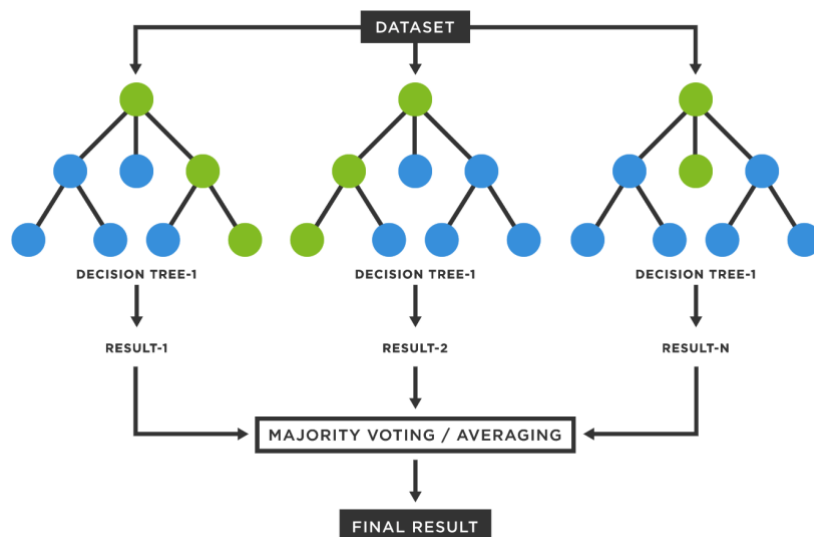


Ilustración 2 Arquitectura de un algoritmo Random Forest

Cuando se utiliza para problemas de clasificación se combinan los árboles de decisión usando la estrategia soft-voting. El clasificador toma las decisiones por votación, seleccionando aquellos árboles en que este más seguro del los resultados. Para problemas de regresión se devuelve la media aritmética o promedio de los árboles de individuales.

Ventajas del algoritmo

La estructura tipo diagrama de flujo imita el pensamiento a nivel humano lo que lo hace muy fácil de entender y interpretar. Además requiere de muy poca preparación de los datos a diferencia de otros algoritmos que requieren una normalización o uso de variables ficticias y puede manejar tanto datos numéricos como categóricos.

Limitaciones del algoritmo

Los árboles de decisión tienen tendencia a sobre ajustarse (overfitting), es decir el modelo tiende a ajustarse demasiado bien a los datos de entrenamiento y como resultado no puede predecir los datos de prueba desconocidos ya que no tiene capacidad de generalizar. Una buena forma de mejorarlo es usando la regulación para mejorar la capacidad de generalizar y combinar los distintos árboles.

Arquitectura

Los bosques aleatorios construyen muchos árboles de decisiones individuales durante el entrenamiento. Las predicciones los árboles se combinan para hacer la predicción final. Utilizando la moda para la clasificación o la media probabilista para la regresión. Dado que utilizan una colección de resultados para tomar una decisión final, se denominan técnicas de conjunto.

La importancia de un nodo se calcula como la disminución de la impureza del nodo ponderada por la probabilidad de alcanzar ese nodo. La probabilidad del nodo se puede calcular por el número de muestras que llegan al nodo, dividido por el número total de muestras. Cuanto mayor sea el valor, más importante será la función.

Para cada árbol de decisión, se calcula **importancia de un nodo** utilizando "Gini Importance", asumiendo solo dos nodos secundarios (árbol binario):

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)}$$

ni_j = la importancia del nodo j

$w_{(j)}$ = número ponderado de muestras que llegan al nodo j

$C_{(j)}$ = el valor de impureza del nodo j

$left(j)$ = nodo hijo de la división izquierda en el nodo j

$right(j)$ = nodo hijo de la división derecha en el nodo j

La **importancia de cada característica** en un árbol de decisiones se calcula como:

$$f_i = \frac{\sum_{j: \text{node } j \text{ splits on feature } i} n_{ij}}{\sum_{k \in \text{all nodes}} n_{ik}}$$

f_i = la importancia de la característica i

n_{ij} = la importancia del nodo j

Se normaliza a un valor entre 0 y 1 dividiendo por la suma de todos los valores de importancia de cada característica.

$$\text{norm}f_i = \frac{f_i}{\sum_{j \in \text{all features}} f_j}$$

Finalmente a importancia de la característica final, a nivel de bosque aleatorio, es su promedio en todos los árboles. La suma del valor de importancia de cada característica en cada árbol se calcula y se divide por el número total de árboles:

$$RF_i = \frac{\sum_{j \in \text{all nodes}} \text{norm}f_{ij}}{T}$$

RF_i = la importancia de la característica i calculada a partir de todos los árboles en el modelo Random Forest

$\text{norm}f_{ij}$ = la importancia de la característica normalizada para i en el árbol j

T = número total de árboles

2.10. Adaptive Neural Fuzzy Interface System (ANFIS)

Un Sistema Neuro-Difuso Adaptativo [18] [19] [20] es una red neuronal artificial cuyas neuronas tienen incorporados el sistema de inferencia difusa de Takagi-Sugeno. Integra tanto las redes neuronales como los principios de lógica difusa. Tienen el potencial de capturar los beneficios de ambos en un solo marco.

El sistema de inferencia corresponde a un conjunto de reglas difusas IF-THEN que tienen la capacidad de aprendizaje para aproximar funciones no lineales. Las redes neuronales aportan un proceso adaptativo, tolerante a fallos y con capacidad de generalizar, por lo que está considerado un estimador universal.

Ventajas del algoritmo

El sistema ANFIS tiene la capacidad de capturar la estructura no lineal de un proceso, la capacidad de adaptación y la capacidad de aprendizaje rápido.

Limitaciones del algoritmo

Las limitaciones son el tipo y número de miembros utilizados en una función, la maldición de la dimensionalidad y el equilibrio entre interpretabilidad y precisión que es considerado uno de los mayores problemas.

Arquitectura

Para simplificar se considera que el sistema de inferencia difusa tiene dos entradas X Y y una salida Z. Para un modelo difuso Takagi-Sugeno de primer orden. El conjunto de reglas difusas IF-THEN para este modelo es:

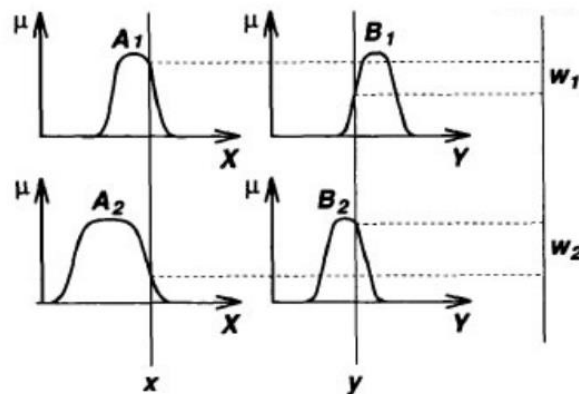


Ilustración 3 Modelo difuso Takagi-Sugeno

Regla 1: si x es A1 e y es B1, entonces $f_1 = p_1x + q_1y + r_1$

Regla 2: si x es A2 e y es B2, entonces $f_2 = p_2x + q_2y + r_2$

Por tanto, la ecuación de Sugeno es la siguiente,

$$f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \bar{w}_1 f_1 + \bar{w}_2 f_2$$

A continuación se muestra la arquitectura completa del modelo ANFIS. Las cinco capas de esta arquitectura son: Fuzzification, Fuzzy Rule, Normalization, Output Membership, Defuzzification.

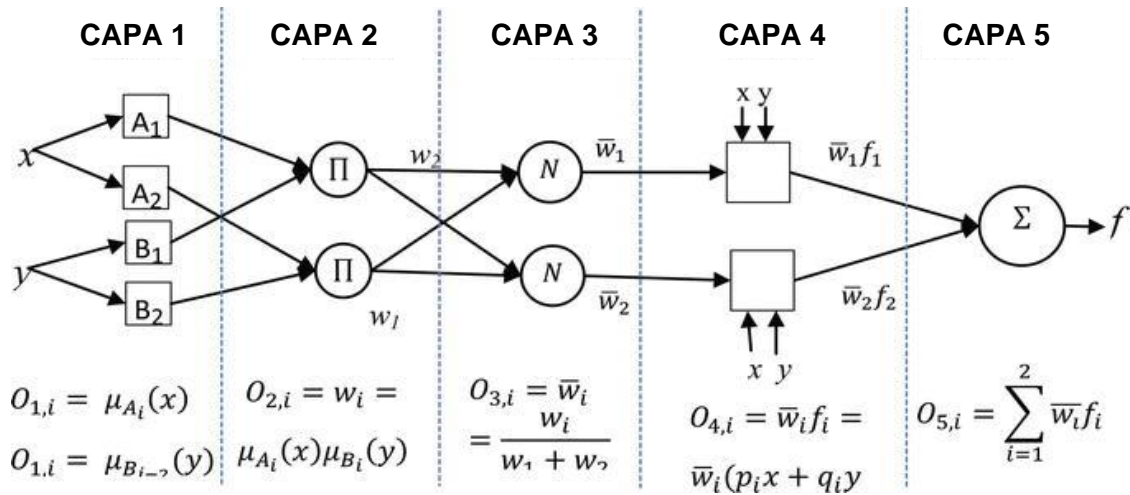


Ilustración 4 Arquitectura completa del modelo ANFIS

Capa 1 – Fuzzification:

La fuzzificación es el proceso de descomponer una entrada del sistema en uno o más conjuntos difusos. El primer paso consiste en tomar las entradas y determinar el grado lingüístico al que pertenecen cada uno de los conjuntos. La función del nodo viene dada por la siguiente función:

$$O_{1,i} = \mu_{A_i}(x), \text{ for } i = 1,2 \text{ or}$$

$$O_{1,i} = \mu_{B_{i-2}}(y), \text{ for } i = 3,4$$

x, y = entradas a la neurona

A_i, B_i = etiquetas lingüísticas

Capa 2 – Fuzzy Rule:

En esta capa los nodos son fijos y su salida representan el producto Π de las señales entrantes que le llegan. La salida de cada nodo de la capa 2 se denomina fuerza de disparo. Cada capa recibe los valores de entrada de la primera y se convierte en una función de pertenencia que representa los conjuntos difusos de las respectivas variables de entrada.

$$O_{2,i} = w_i = \mu_{A_i}(x)\mu_{B_i}(y), i = 1, 2$$

Capa 3 - Normalization

Los nodos de esta capa también son fijos, cada nodo normaliza la fuerza de disparo de una regla de la capa anterior y calcula la relación entre la fuerza de disparo de la i -enésima regla y la suma de las fuerzas de disparo de todas las reglas.

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}$$

Capa 4 - Output Membership

Los nodos de esta capa se adaptan a la función del nodo. Donde w es la fuerza de disparo normalizada y $\{p_i, q_i, r_i\}$ es un polinomio de primer orden. $O_{4,i}$, representa la salida de la capa 4. Los parámetros de esta capa son parámetros lineales conocidos como parámetros consecuentes. Estos parámetros se identifican durante el proceso de entrenamiento del modelo ANFIS.

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i x + r_i), i = 1,2$$

Capa 5 - Defuzzification

Este último nodo se denomina capa de salida y se etiqueta como " Σ ". Esta capa suma las salidas de todas las reglas de la capa anterior y convierte el resultado difuso en una salida nítida.

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}$$

2.11. Support Vector machine (SVM)

Las Máquinas de Soporte Vectorial [21] [22] [23] son un conjunto de métodos de aprendizaje supervisado que se utilizan para la clasificación, regresión y detección de valores atípicos. Dado un conjunto de datos de entrenamiento el modelo es capaz de etiquetar las clases y construir un modelo que prediga la clase a la que pertenece la nueva muestra.

Una SVM es un modelo que representa los puntos de muestra en el espacio, separando las clases a espacios más amplios posibles mediante un hiperplano de separación. El vector entre los 2 puntos de las clases más cercanas se llama vector de soporte. Cuando se introduce una nueva muestra en función de los espacios a los que pertenece se clasifica en una clase o otra. Una buena separación de clases permite una clasificación óptima.

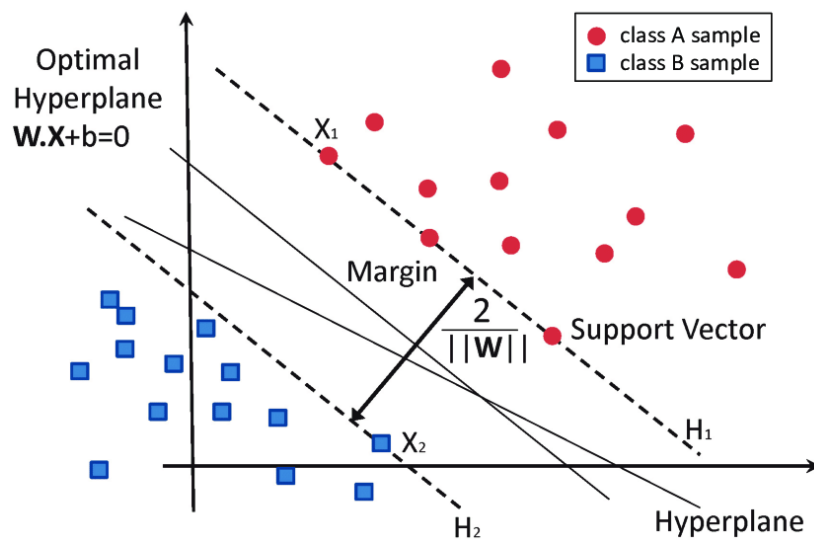


Ilustración 5 Ejemplo de separación de clases mediante el vector soporte

Ventajas del algoritmo

Es muy efectivo con datasets de gran dimensionalidad, es eficaz incluso cuando el número de dimensiones supera el número de muestras. Utiliza un subconjunto de puntos de entrenamiento en la toma de decisiones, lo que lo hace muy eficiente en memoria. Además es muy versátil ya que se pueden especificar diferentes funciones del núcleo para la función de decisión.

Limitaciones del algoritmo

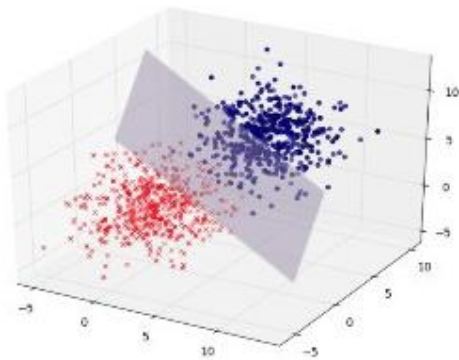
Las SVM no proporcionan estimaciones de probabilidad, se calculan mediante validación cruzada de cinco veces lo que lo hace muy costosa a nivel computacional.

Arquitectura

Una SVM realiza la clasificación para encontrar el hiperplano que maximiza el margen entre las dos clases cuyo límite de decisión esté más alejado de cualquier punto de datos. Los vectores que definen el hiperplano son los vectores de soporte. El hiperplano separador se encuentra en los puntos de datos más cercanos al límite. Y estos puntos se denominan vectores de soporte.

Ecuación del hiperplano

$$w^T x = 0$$



Ecuación de línea

$$y = ax + b$$

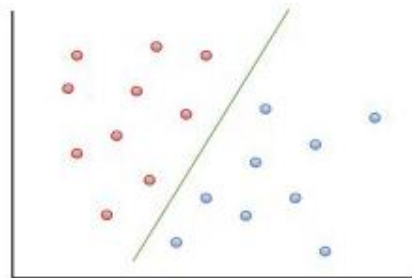


Ilustración 6 Representación de la Ecuación del Hiperplano y Ecuación de Línea

El hiperplano óptimo se encuentra representado en la línea amarilla ya que es el punto donde se maximiza el margen. Si se divide la distancia entre hiperplanos y dividimos entre dos obtenemos el límite de decisión. Para maximizar el margen si se consideran dos dimensiones, se obtiene la siguiente ecuación para la hiperlínea.

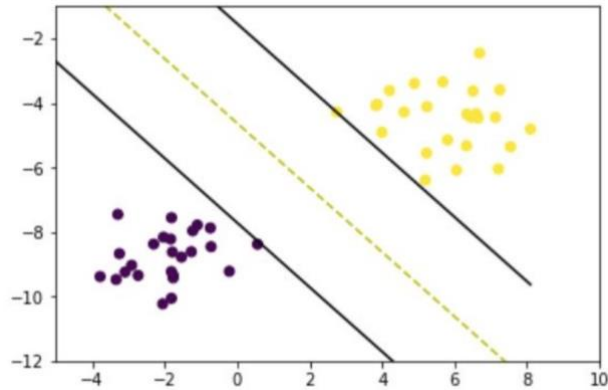


Ilustración 7 Muestra del hiperplano óptimo

$$w \cdot x_i + b \geq 1 \text{ para } x_i \text{ siendo la clase 1}$$

$$w \cdot x_i + b \leq -1 \text{ para } x_i \text{ siendo la clase -1}$$

- Si $w \cdot x + b = 0$, entonces obtenemos el límite de decisión.
- Si $w \cdot x + b = 1$, entonces obtenemos la clase hiperplano +
- Si $w \cdot x + b = -1$, entonces obtenemos la clase hiperplano -

Se obtiene el siguiente resultado, como se puede observar es necesario elegir el margen máximo uno entre ellos.

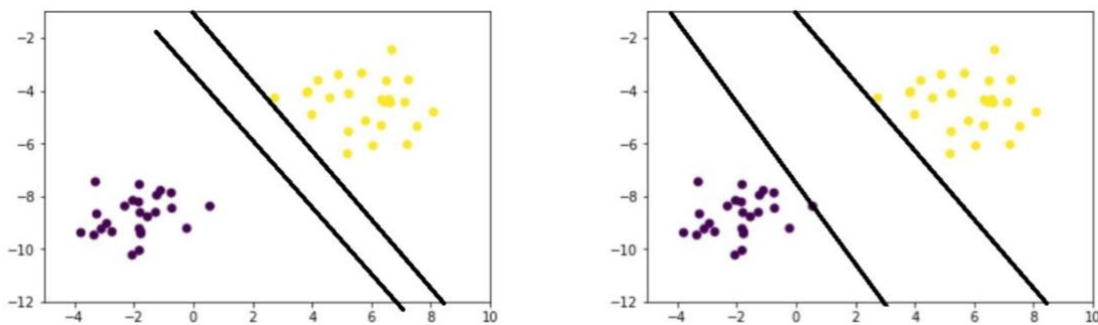


Ilustración 8 Representación gráfica de los límites máximos de los hiperplanos

Elegimos el que tenga la magnitud mínima de w

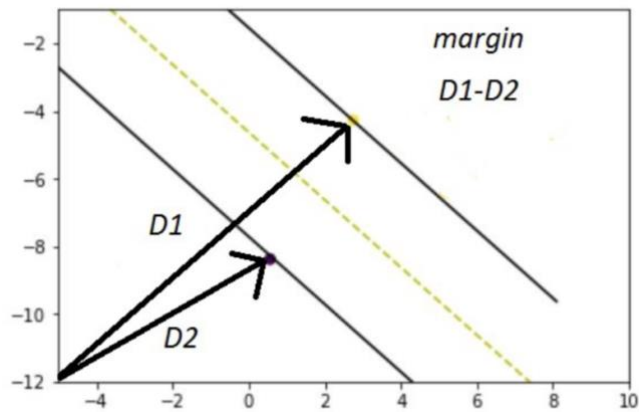


Ilustración 9 Representación de los márgenes $D1$ y $D2$

$$D1 = w^T x + b = 1$$

$$D2 = w^T x + b = -1$$

$$w^T x + b - 1 = 0$$

$$w^T x + b - 1 = 0$$

Si se soluciona de forma algebraica obtenemos:

$$\min \frac{1}{2} \|w\|^2$$

El atractivo de la SVM es que si los datos se pueden separar linealmente, existe un valor mínimo global único.

2.12. Feedforward neural network (FNN)

Una red neuronal de retroalimentación [24] [25] [26] [27] es el tipo de red neuronal más sencilla, consta de una sola capa oculta y una capa de salida. La información se mueve en una única dirección, de los nodos de entrada a través de los nodos ocultos hasta llegar a los nodos de salida. No hay retroalimentación entre capas por lo que se les denomina redes neuronales Feedforward.

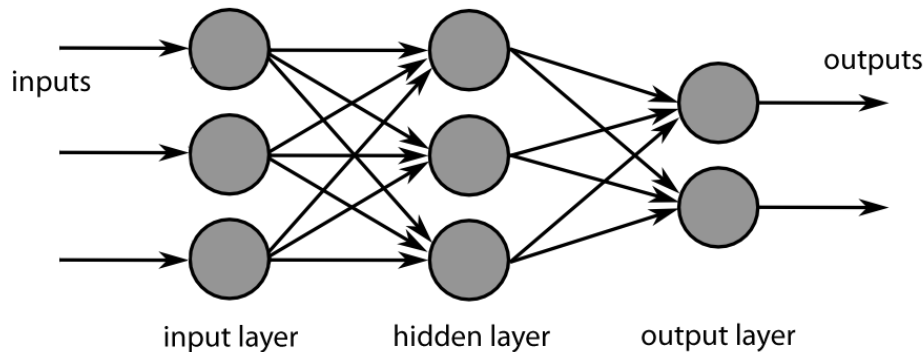


Ilustración 10 Arquitectura de una Red Neuronal de Retroalimentación

El tipo más simple de red neuronal Feedforward es el perceptrón, la salida se calcula directamente a partir una serie de ponderaciones, la suma del producto de los pesos de las ponderaciones y las entradas se calcula en cada nodo. Si el valor es superior a un umbral (mayor que 0) la neurona se dispara y toma el valor de activación (igual que 1), en caso contrario toma el valor de desactivación (igual que -1).

Ventajas del algoritmo

Una de las mayores ventajas de este algoritmo es que requiere poco esfuerzo computacional y pocos uso de memoria para almacenar los pesos. Además, tienen gran capacidad de generalizar y diferencia de otro tipo de redes neuronales son las más simple lo que las convierte en más fáciles de comprender. Se utilizan principalmente en los casos en que los datos que se van a aprender no son secuenciales ni dependen del tiempo.

Limitaciones del algoritmo

Las redes neuronales son sensibles a la aleatorización inicial de sus matrices de peso lo que puede provocar que la red pueda terminar atrapada en un mínimo local y es posible que nunca pueda aumentar su precisión por encima de un cierto umbral. Requiere la identificación de todas las posibles perturbaciones y su medición directa ya que no puede hacer frente aquellas perturbaciones que no haya medido anteriormente.

Arquitectura

Las redes neuronales están basadas en la comprensión biológica de las neuronas de nuestro cerebro. Una neurona lo que hace es recibir información de otras neuronas, procesar la información y enviar el resultado a otras neuronas. Esta comprensión biológica se puede traducir en un modelo matemático como se muestra a continuación.

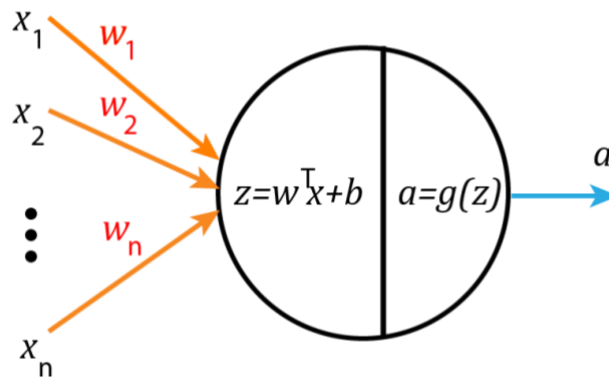


Ilustración 11 Modelo matemático de un perceptrón

La neurona toma un vector de características de entrada X_1, X_2, \dots, X_n , cada uno de ellos se multiplica por un peso específico, W_1, W_2, \dots, W_n . Las entradas ponderadas se suman y se les agrega un valor constante llamado sesgo para producir la entrada neta de la neurona.

$$z = \sum_{i=1}^n x_i w_i + b$$

La entrada luego pasa a través de una función de activación g para producir la salida que luego se transmite a otras neuronas.

$$a = g(z) = g\left(\sum_{i=1}^n x_i w_i + b\right)$$

La función de activación es elegida en el diseño y w_i b se ajustan durante el proceso de entrenamiento de la red neuronal.

Existen diferentes funciones de activación que puede usar en una red neuronal. A continuación se muestra una tabla con las más importantes.

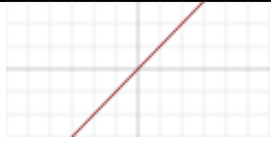


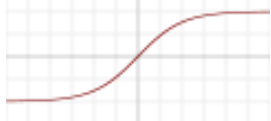

Nombre	Gráfico	Función $f(x)$	Derivada $f'(x)$	Rango
Identity		x	1	$(-\infty, \infty)$
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$\begin{cases} 0 & \text{if } x < 0 \\ \text{undefined} & \text{if } x \geq 0 \end{cases}$	$\{0,1\}$
Logistic, sigmoid		$\sigma = \frac{1}{1 + e^{-x}}$	$f(x)(1 - f(x))$	$(0,1)$
Hyperbolic tangent (tanh)		$\begin{aligned} \tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \end{aligned}$	$1 - f(x)^2$	$(-1,1)$
Rectified linear unit (ReLU)		$\begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$	$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$[0, \infty)$

Tabla 4 Funciones de activación de una red neuronal

El uso de cualquiera de estas funciones asegura que la salida sea un número continuo (a excepción del binario) y, por lo tanto, no todas las funciones de activación producen un clasificador lineal.

En el caso del perceptrón utiliza la función de activación binaria, que cambia los pesos de la red en función de la derivada de la función de activación, que es cero. Por lo que, no hay actualizaciones de las ponderaciones en la propagación hacia atrás.

2.13. Value Investing aplicando técnicas de aprendizaje automático

En esta sección se ofrece una descripción general de diferentes trabajos llevados a cabo sobre la inversión basada en el análisis fundamental abordado desde el punto de vista del machine learning. Los artículos en revisión discuten sobre los enfoques empleados para realizar una predicción del precio de las acciones de una compañía.

Investigación 1

Los autores Anita H. y S.F. Rodd en su trabajo de investigación [28] proponen utilizar un Red Neuronal feedforward para realizar el análisis fundamental y predecir el desempeño de una compañía. En una primera fase utilizan ratios financieros como el ROCE, Deuda/Recursos propios ratio, crecimiento de las ventas y el rendimiento de dividendos, para clasificar las compañías según si su desempeño es bueno o malo (0,1).

En la segunda fase utilizan un algoritmo de clasificación como el Support Vector Machine para comparar los resultados y encontrar los ratios que mayor peso tienen en la predicción.

Investigación 2

En este artículo de investigación [29] Yuxuan Huang plantea construir una cartera de acciones mediante un ranking. Tratando de predecir el rendimiento relativo trimestral de cada una de las 70 acciones extraídas del índice S&P100 (100 empresas de mayor capitalización de EEUU) haciendo uso de ratios como (ROA, deuda, margen bruto, rotación de activos).

La cartera se divide en acciones de "compra" y "venta" en las que se identifican las acciones con peor desempeño, ya que si se disminuyen estas acciones se puede generar mayores ganancias. La puntuación de la cartera tiene en cuenta no solo su rendimiento, sino también su desviación estándar durante el período de validación.

En el desarrollo experimental utiliza tres algoritmos de aprendizaje automático, los RF, ANFIS y FNN. Cada algoritmo se valida en las dos carteras de compra y venta, en las que obtiene un mayor rendimiento con los RF. Aplicando la selección de funciones y agregando los diferentes algoritmos, el modelo agregado alcanza una puntuación de cartera de 0,759 para compra y -0,335 para venta llegando a superar el índice de referencia del mercado bursátil.

Investigación 3

En este artículo de investigación [30] los autores plantean un enfoque muy parecido a la anterior investigación. Utilizan como variable objetivo a predecir el rendimiento relativo trimestral de una acción con respecto al promedio industrial Dow Jones (DJIA).

El rendimiento relativo de una acción es la diferencia entre su rendimiento absoluto y el rendimiento de algún índice de referencia. Al restar el desempeño general del mercado del desempeño de cada acción individual, podemos filtrar los factores que afectan al mercado en general. Utilizando esta técnica de este tipo ayuda a reducir la complejidad del problema de predicción y mejora el rendimiento de la predicción.

Para construir la cartera analiza 70 acciones las cuales se clasifican según su rendimiento previsto. Las 30 acciones con mejores rendimiento relativos previstos, se utilizan para construir la cartera de compra, y las 30 acciones más bajas se colocan en la cartera de venta.

Los modelos de aprendizaje automático utilizados para las predicciones son los modelos FNN y ANFIS. Obteniendo mejores resultados en las carteras de compra y mejor rendimiento con el modelo FNN.

Capítulo 3. DESARROLLO DEL SISTEMA

3.1. Introducción

El objetivo que persigue este proyecto es construir una herramienta de apoyo para pequeños inversores, que les ayude a mejorar el rendimiento de sus carteras mediante el uso de predicativa analítica y obtener nuevas ideas de inversión. A través de una interfaz gráfica el usuario debe ser capaz de entender de forma rápida y sencilla que ocurre en su cartera y si debe realizar algún cambio en su estrategia o existe alguna oportunidad nueva en el mercado

De este modo, este proyecto se ha enfocado en diferentes fases: extracción de datos, almacenaje, algoritmo de aprendizaje automático y visualización de los datos.

En la fase de extracción de los datos se realizará un *scraping* de los datos financieros de más de 100 compañías que cotizan en las bolsas americanas, con un especial foco en las empresas de SP500, y además de la obtención de datos del bróker DEGIRO [31] para mantener actualizado el sistema con los últimos movimientos que se realicen en la cartera.

La fase de almacenaje tratará todos los aspectos relacionados con el guardado de los datos obtenidos. Para este propósito, se ha elegido Elasticsearch [32] por su entorno y herramientas que facilitarán la posterior visualización de datos.

La fase del algoritmo aprendizaje automático se generará un dataset con todas los ratios más relevantes de todas las compañías y se analizará el poder predictivo de cada una de ellas, para luego finalmente entrenar un algoritmo y integrarlo en nuestro sistema.

Finalmente, la fase de visualización se desarrollará un dashboard intuitivo con Kibana Elasticsearch que ayude al inversor a ver las posiciones actuales de su cartera, una predicción que indicará si comprar o vender las acciones que actualmente tiene en cartera, y finalmente un apartado de sugerencias con empresas que el algoritmo determine que se encuentran infravaloradas y tienen un buen potencial de rendimiento.

3.2. Proceso de extracción de los datos

Para posteriormente poder aplicar un algoritmo de aprendizaje automático es muy importante obtener datos de calidad, es por ello que se ha elegido la página *app.tikr.com*. Se trata de una página donde podemos ver todos los ratios financieros de multitud de compañías de los mercados bursátiles más importantes, especialmente el mercado americano y europeo.

Su principal base de datos proviene de una empresa especializada en datos y noticias financieras, se trata de Reuters desde su filial Refinitiv [33]. La ventaja de utilizar TIKR es que actualmente la página se encuentra en fase beta por lo que es totalmente gratuita y podemos acceder a todos los datos y servicios.

Para el desarrollo del proyecto también se ha estimado la opción de contratar los servicios de Refinitiv pero el precio es muy costoso sólo apto para grandes empresas.

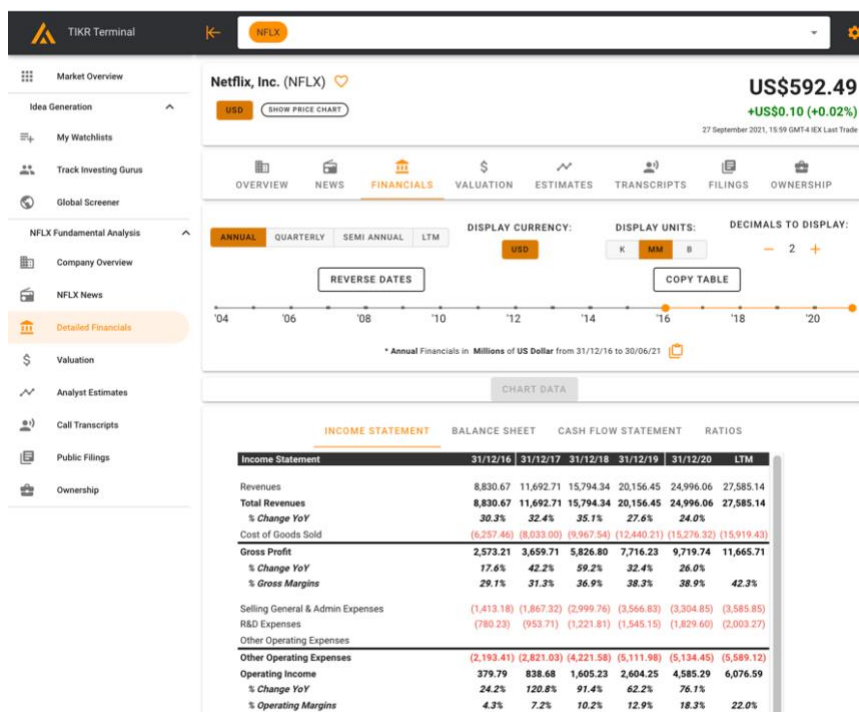


Ilustración 12 Ratios financieros en *app.tikr.com*

Para realizar el proceso de extracción de los datos, la página no dispone de una API pública con la que se puedan obtener los datos que muestran en su página. Por ese motivo se ha realizado un *sniffing* de las conexiones que realiza el navegador con el servidor para tratar de reproducir el mismo entorno en un script de Python.

La ventaja principal de utilizar un script de Python es poder mantener al día nuestro sistema, es muy importante que cada vez que se publiquen nuevos reportes financieros o hayan cambios en los precios nuestro sistema se actualice de forma automática, ejecutando el algoritmo de aprendizaje automático y subiendo los datos a la base de datos de Elasticsearch.

3.2.1. Web Scraping

Como ya se ha comentado en el apartado anterior, se ha realizado un estudio de las peticiones que realiza el navegador con el servidor para su posterior integración en un script de Python. A continuación se muestra un resumen del funcionamiento de la API REST estudiada.

- **Obtener ID de la compañía**

GET: https://tjpay1dyt8-dsn.algolia.net/1/indexes/tikr-feb/query?	
Descripción: Dado el nombre de una compañía retorna un listado con las empresa que coinciden con el nombre e información básica como el ID, exchange, divisa y sector.	
Parámetros	
Nombre	Descripción
"query"	Nombre de la compañía
"distinct": 2	Valor siempre constante
Respuesta	
Ver fichero: Scraper/Data/cid/cid_21835.json	

Tabla 5 API REST: Buscar compañía y obtener datos básicos

- **Obtener RIC de la compañía**

GET: https://rwrjol17eb.execute-api.us-east-1.amazonaws.com/prod/trkdid	
Descripción: Mediante los datos obtenidos en la API anterior podemos obtener el número RIC de la compañía, que posteriormente se utilizará para obtener la lista de accionistas.	
Parámetros	
Nombre	Descripción
"auth"	Key de autenticación se obtiene al hacer login en la página web
"ticker"	Ticker ID de la empresa (numérico)
"cid"	ID de la empresa (siempre numérico)
"companyname"	Nombre de la compañía
"symbol"	Tickr ID de la empresa (alfabético)
"tid"	Ticker ID de la empresa (numérico)
"exchange"	Nombre del mercado de valores donde cotiza
"primaryTicker": "false"	Valor siempre constante
"jwt"	Key de autenticación el mismo que auth
"from": "ciq2ric"	Valor siempre constante
"v": "v1"	Valor siempre constante
Respuesta	
Retorna un valor numérico llamado RIC que se utilizará en otra API REST	

Tabla 6 API REST: Obtener RIC: Reuters Instrument code

- **Obtener datos financieros**

GET: https://rwrj0l17eb.execute-api.us-east-1.amazonaws.com/prod/fin	
Descripción: Dado el id de una compañía retorna todos los datos financieros codificados (ver fichero adjunto conversion.py)	
Parámetros	
Nombre	Descripción
"auth"	Key de autenticación se obtiene al hacer login en la página web
"cid"	ID de la empresa (siempre numérico)
"p": "1"	Valor siempre constante
"repid": 1	Valor siempre constante
Respuesta	
Ver fichero: Scraper/Data/financials/financials_21835.json	

Tabla 7 API REST: Obtener datos financieros

- **Obtener lista de accionistas**

GET: https://r54xadzxvh.execute-api.us-east-1.amazonaws.com/prod/shareholders	
Descripción: Dado el tiker de una compañía retorna una lista con los accionistas mayoritarios de la compañía, así como también el número de acciones que poseen	
Parámetros	
Nombre	Descripción
"auth"	Key de autenticación se obtiene al hacer login en la página web
"ticker"	Valor RIC
Respuesta	
Scraper/Data/price/price_21835.json	

Tabla 8 API REST: Obtener lista de accionistas

*Los datos aportados por esta API finalmente no se han utilizado ya que solo aporta información del año en curso y no proporciona información histórica.

- **Obtener evolución del precio**

GET: https://rwrj0l17eb.execute-api.us-east-1.amazonaws.com/prod/price	
Descripción: Dado el id y tikr de la empresa retorna el precio histórico de la acción y el volumen de acciones ordenados por fecha.	
Parámetros	
Nombre	Descripción
"auth"	Key de autenticación se obtiene al hacer login en la página web
"cid"	ID de la empresa (siempre numérico)
"currency"	160: USD, 50: EUR (valor numérico)
"tid"	Ticker ID de la empresa
"v":"v0"	Valor siempre constante
Respuesta	
Scraper/Data/insiders/insiders_21835.json	

Tabla 9 API REST: Obtener precio histórico

3.2.2. Desarrollo del script de scraping

Una vez obtenida toda la información de cómo funcionan las peticiones del servidor se ha integrado en un script de Python **Anexo 1**. En el siguiente esquema se muestra como se ha implementado la API para obtener todos los datos financieros.

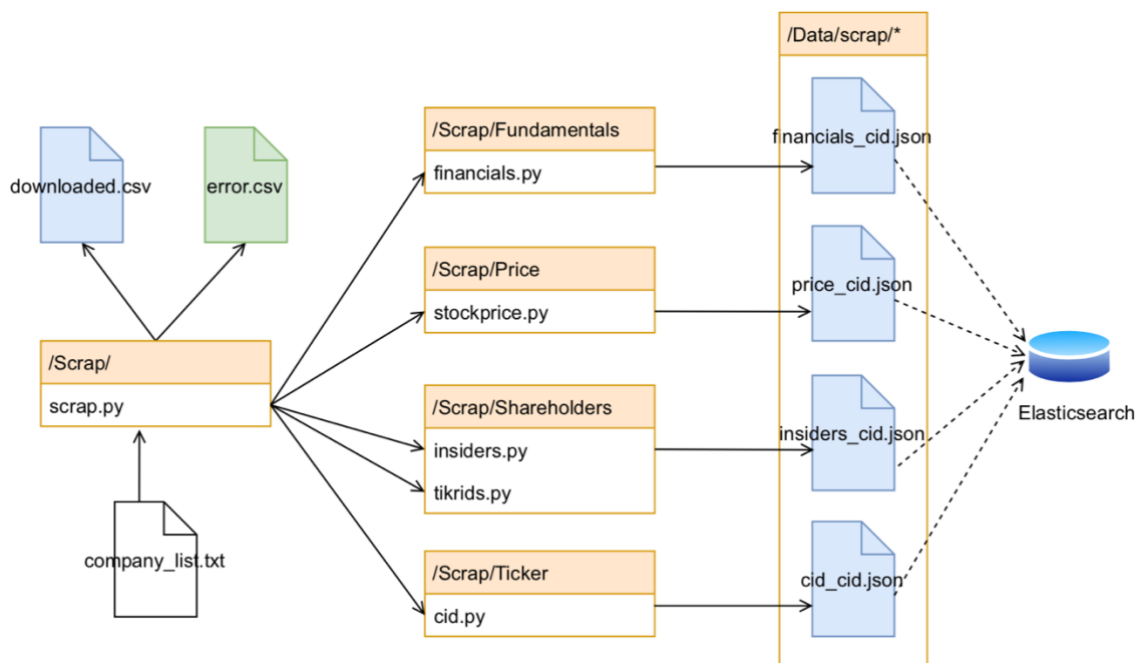


Tabla 10 Arquitectura del script de scraping

El fichero principal es **scrap.py** que hace las llamadas a los distintos ficheros de Python ubicados en carpetas. El script empieza leyendo el fichero `company_list.txt` donde se encuentra un listado con el nombre de más de 100 empresas. Después, lee la key de autenticación que previamente hemos extraído de la página web `app.tkr.com` al hacer un login con nuestros credenciales.

El script ejecuta el fichero **cid.py** para obtener la información básica de la compañía, que después más tarde se almacena en un fichero llamado **cid_xxx.json** donde xxx es el company id. Una vez obtiene la información básica se ejecuta **financials.py** que obtiene los datos financieros y los almacena en un fichero llamado **financials_xxx.json**.

En el siguiente paso se ejecuta **tikrids.py** para obtener el RIC “Reuters Instrument Code” necesario para ejecutar **insiders.py**, en este último se obtiene el listado de accionista y los almacena en el fichero **insiders_xxx.json**.

Finalmente, se ejecuta **stockprice.py** para obtener el precio histórico que se almacena en el fichero **price_xxx.json**. Si surge algún problema en alguna de las peticiones que realiza en el servidor, el script continua con el siguiente nombre de la lista y genera un log en el fichero `error.csv` con información del problema ocurrido. Aquellas empresas que se descargan correctamente se almacena su nombre y company id en el fichero `downloaded.csv`.

Los datos financieros obtenidos se guardan tanto de forma local como en el servidor de Elasticsearch para luego ser procesados y extraer la información necesaria para generar el dataset y aplicar el algoritmo de Machine Learning.

3.2.3. Creación del dataset

Con los datos obtenidos con el script de scraping **Anexo 1** se realiza una extracción de los ratios financieros que pueden tener más poder predictivo a la hora de predecir el precio de la acción. Para esta tarea se ha creado un script llamado `dataset_creator.py` que genera el `dataset.csv`

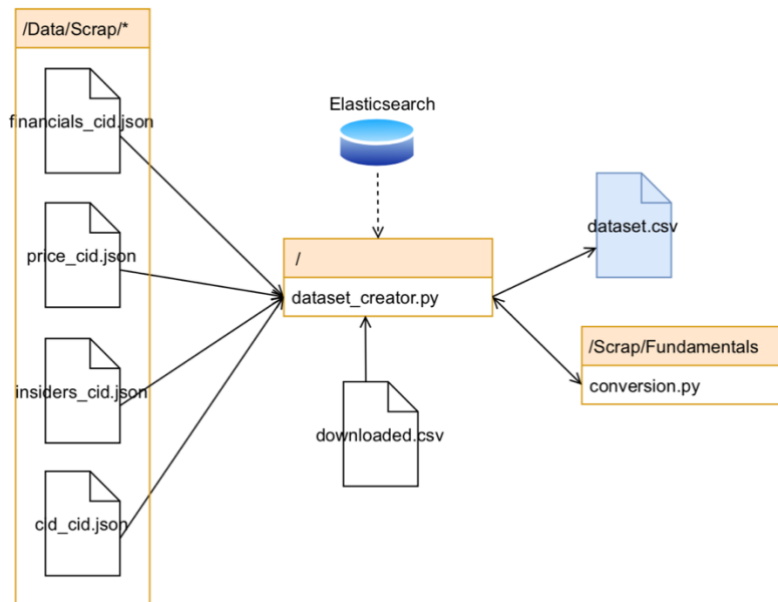


Ilustración 13 Arquitectura del script `dataset_extractor`

El script **dataset_creator.py** lee el fichero `downloaded.csv` que contiene el listado de todas las empresas que previamente se han descargado con su número cid. El script puede acceder tanto a los datos alojados en el servidor, como de forma local de la carpeta Data.

Para empezar el script lee el fichero `cid_cid.json` que contiene información básica como el sector, moneda local, nombre de la empresa y mercado bursátil. Después sigue con el fichero **insiders_cid.json**, en este fichero revisa los 10 inversores con mayor participación y realiza un ponderación para comprobar si mayoritariamente están comprando o vendiendo sus acciones y en que magnitud lo están haciendo.

Con el fichero **price_cid.json** se extrae el precio de cierre de cada año, como este proyecto está enfocado a predecir el precio, el precio asignado siempre será el del año que le precede. Es decir los ratios financieros del año 2015, se le asignará el precio que tuvo un año después, es decir 2016.

A continuación lee el fichero **financials_cid.json**, en este fichero se encuentran todos los ratios financieros codificados con un valor numérico. Para traducir los ratios de interés, es necesario ejecutar **conversión.py**, se trata de un diccionario con el nombre del ratio y el número que le corresponde. Esta traducción se descubrió en un fichero javascript de la página de Tigr que ellos utilizan para parsear los datos.

Los ratios seleccionados para formar el dataset son los siguientes: Depreciation & Amortization, Operating Income, Interest Expense, Income Tax Expense, Weighted Average Diluted Shares

Outstanding, Capital Expenditure, Cash And Equivalents, Current Portion of Long-Term Debt, Long-Term Debt, Goodwill, Total Equity, Cash Flow per Share.

3.2.4. Obtener datos del bróker

En los últimos años se ha popularizado el bróker DEGIRO [31] en mayor parte por sus bajas comisiones a la hora de comprar acciones. Pequeños inversores lo utilizan diariamente para realizar inversiones a largo plazo.

Para este proyecto, se obtendrán principalmente las posiciones actuales de la cartera, número de acciones, precio de la acción y los movimientos a lo largo del tiempo, que se utilizarán para poder ver más adelante su evolución, en relación a un índice de referencia que en este caso se ha elegido el Standard & Poor's.



En este caso para obtener los datos del bróker no existe una API oficial, pero en Github hay algunos proyectos no oficiales muy básicos para poder realizar la conexión que facilitará el acceso. Concretamente para este desarrollo se ha inspirado en uno de los repositorios [34].

3.2.5. Obtener posiciones del bróker

Uno de los elementos más importantes para invertir en el mercado de valores, es tener una visión de conjunto de nuestras posiciones. Para ello se ha desarrollado un script para descargar el listado de acciones en posesión, número de acciones, precio actual y evolución de la cartera a lo largo del tiempo.

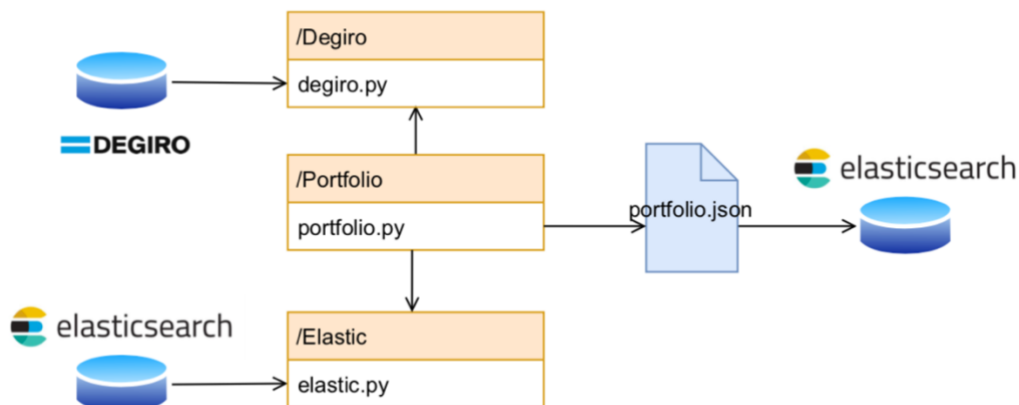


Ilustración 14 Arquitectura del script portfolio

El fichero principal es **portfolio.py**, que llama a **degiro.py**, este último contiene todas las funciones para conectarnos con el servidor de DEGIRO **Anexo 1**. Primero se encarga de realizar las peticiones de login con los credenciales del fichero config.json y luego obtiene los datos con

la información de nuestro portfolio codificada, número de acciones en cartera, id de la empresa, precio de cierre. El script aplica un filtrado y elimina las posiciones que ya no tenemos en cartera, con los id de empresa obtenidos llama a otra API que retorna los nombres de las compañías.

Con toda la información, se genera un fichero JSON con un resumen de toda la cartera y se cumplimenta con las predicciones hechas por el algoritmo de machine learning que previamente se han alojado en Elasticsearch. Para ello se utiliza una librería de Python específica para Elastic [35] que se accede desde el *wrapper elastic.py*.

Las predicciones que obtenemos son en formato precio, por lo que es necesario hacer una traducción a cuál es la acción que debemos realizar (comprar, vender, mantener). Utilizando el precio de cierre actual de la compañía y el precio predicho se calcula el margen de seguridad porcentual y se determina la acción según la siguiente tabla:

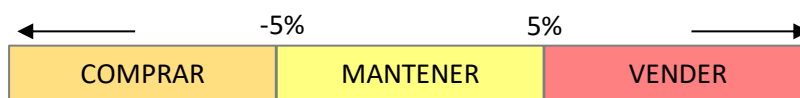


Ilustración 15 Recomendaciones según el margen de seguridad

Si el precio de la acción fuera 60€ y el precio predicho 70€ el margen sería del -17% por lo que el sistema indicaría que podemos comprar. En caso que el margen se encuentre entre el -5% y 5% indicará que es mejor mantener posiciones y si supera el 5% recomendará vender.

En cuanto a la estructura que se ha aplicado al fichero JSON es la siguiente:

Nombre	Descripción
id	Id interno de la empresa
name	Nombre de la empresa
country	País de donde provienen las acciones
currency	Moneda de las acciones (USD, EUR, GB)
symbol	Símbolo de la empresa
closePrice	Precio de cierre
status	Predicción puede ser sell, buy o hold
size	Número de acciones en posesión
value	Número total de acciones * Precio en EUR de la acción
plBase	Precio base inicial desde la primera compra de acciones
growth	Crecimiento absoluto desde la primera compra de acciones
growthPct	Crecimiento porcentual
breakEvenPrice	Precio medio obtenido por las distintas compras de acciones

Tabla 11 Estructura del fichero portfolio.json

Una vez generado el fichero se almacena de forma local y además se envía al servidor de Elasticsearch para posteriormente poderlo visualizar con Kibana.

3.2.6. Obtener rendimiento de la cartera

El rendimiento de la cartera es una referencia que permite evaluar si ganamos o perdemos con nuestra inversión a lo largo del tiempo. Normalmente se compara el rendimiento contra un índice de referencia, lo más común es utilizar el índice Standard & Poor's 500, también conocido por SP500. Este índice se basa en la capitalización bursátil de las 500 empresas más importantes de las bolsas americanas el NYSE o NASDAQ. En la siguiente ilustración se muestra un ejemplo del rendimiento de una cartera vs el SP500.

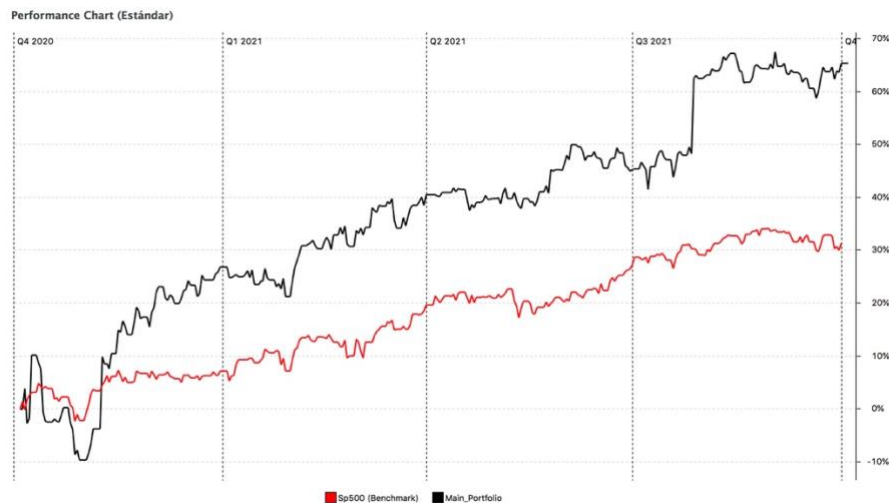


Ilustración 16 Rendimiento cartera vs SP500

Para obtener el rendimiento de la cartera se ha construido el script **performance.py** que obtiene toda la información de DEGIRO **Anexo 1**. Para empezar descarga todos los movimientos de nuestra cartera, tanto de compra como de venta. Después genera un listado con el valor diario de nuestras posiciones realizando una conversión a euros cuando se trata de una moneda distinta al euro. Una vez obtenido el valor diario se calcula la variación porcentual acumulada de nuestro portfolio y se realiza la misma operación sobre el índice del SP500.

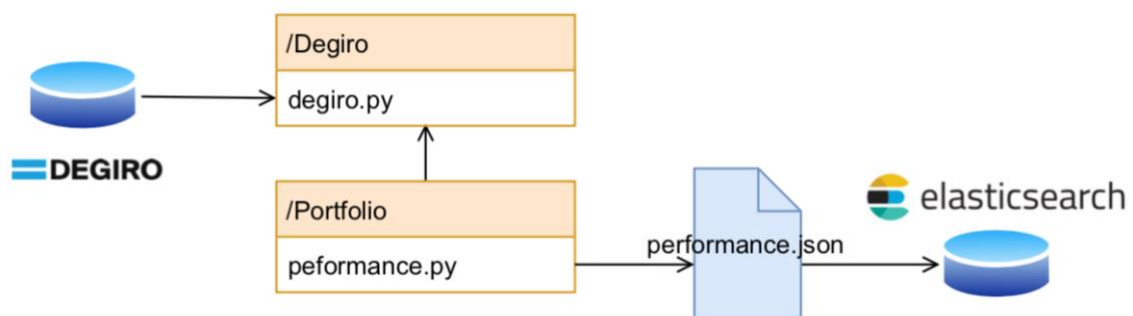


Ilustración 17 Arquitectura del script performance

Las variaciones porcentuales de ambos obtenidas, junto con su fecha se añade en el fichero **performance.json** que después se indexa en servidor de Elasticsearch.

En cuanto a la estructura que se ha aplicado al fichero JSON es la siguiente:

Nombre	Descripción
p	Variación porcentual del precio de la cartera
s	Variación porcentual del precio del SP500
d	Fecha en formato ISO 8601

3.3. La base de datos

Para el almacenaje de los datos se ha elegido **Elasticsearch** [32] que permite almacenar, buscar y analizar datos estructurados y no estructurados. Además de ser un sistema muy flexible y escalable. Su motor de búsqueda está basado en Apache Lucene que permite realizar búsquedas de texto en entornos Big Data.

El software es gratuito y abierto para cualquier tipo de datos, tanto numéricos, textuales o geoespaciales. Además, es conocido por su API REST disponible para múltiples lenguajes, entre ellos Python, permitiendo administrar, consultar e indexar datos de forma rápida y segura.

Elasticsearch forma parte como un componente más de **Elastic Stack**, que es un conjunto de herramientas de análisis de datos, almacenamiento, visualización con el módulo **Kibana** y el procesamiento y transformación de datos con **Logstash**.

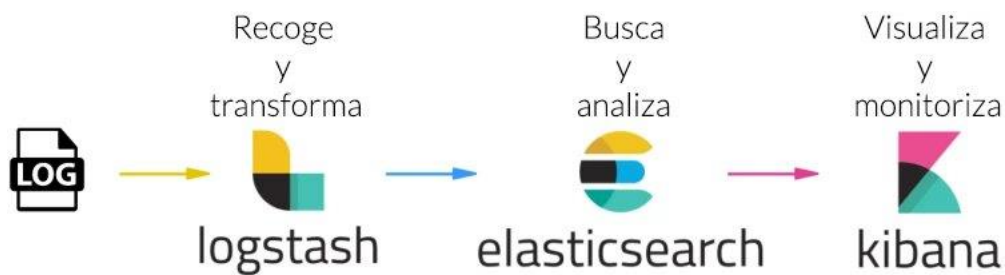


Ilustración 18 Componentes de Elastic Stack

Se usa para búsquedas en sitios webs o aplicaciones, monitoreo y rendimiento de aplicaciones, métricas de infraestructura y **análisis de negocios** que es la parte que nos interesa para el desarrollo de este proyecto.

Los datos se almacenan como documentos JSON. Una colección de documentos relacionados entre sí se denomina **índice**. Elastic utiliza una estructura de índices invertidos que permite identificar cada palabra y ofrecer una búsqueda de datos casi instantánea.

Es un sistema distribuido y por lo tanto los documentos almacenados se van distribuyendo en distintos contenedores llamados **shards**, los shards están duplicados por lo que ofrecen copias redundantes de los datos a lo largo de todos los servidores permitiendo escalar el sistema de forma horizontal.

3.3.1. Configuración del sistema

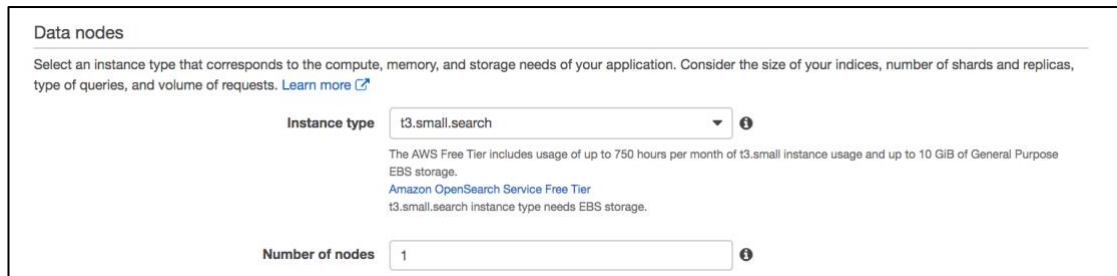
En este caso se ha elegido un servicio en la nube que ofrece Amazon Web Services (AWS) llamado **OpenSearch Service** [36]. Se trata de un servicio distribuido de análisis y búsqueda de datos en código abierto para entornos Big Data.



OpenSearch es el sucesor del servicio Amazon Elasticsearch Service que se desarrolló debido al cambio de políticas de licencias de Elastic NV. En lugar de utilizar la licencia permisiva ALv2 decidieron lanzar las nuevas versiones de Elasticsearch y Kibana bajo su propia licencia, lo que llevó a Amazon crear y mantener la última versión ALv2 de Kibana y Elasticsearch bajo el nombre de OpenSearch.

Desde los servicios AWS podemos elegir entre OpenSearch o las versiones anteriores de Elastic bajo la licencia ALv2, para este proyecto se ha elegido la última versión de Elastic v7.10.2.

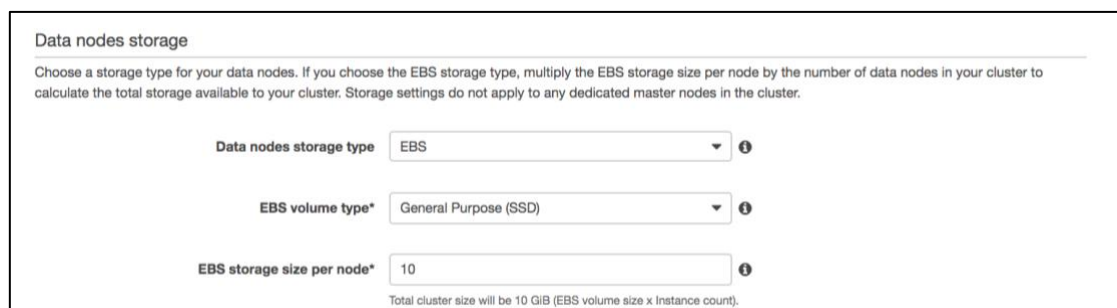
Para la configuración de los nodos se ha seleccionado una del tipo `t3.small.search` ya que ofrecen 750 horas gratuitas de instancias básicas para trabajos de prueba y además incluye 10GB de almacenamiento por clúster.



The screenshot shows the 'Data nodes' configuration page in the AWS console. It includes a dropdown menu for 'Instance type' set to 't3.small.search' and a text input for 'Number of nodes' set to '1'. There is also a small informational text block about the AWS Free Tier and EBS storage requirements for the selected instance type.

Ilustración 19 Configuración AWS: Tipo de instancia

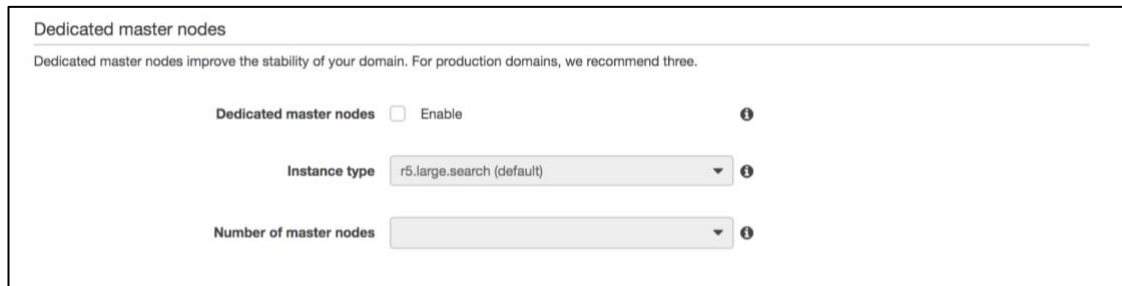
El servidor se encuentra alojado en París por lo que ofrece una baja latencia, además de ser el servidor con el precio por hora más barato 0,038USD/hora. Ofrecen 2GB de RAM y 2 CPU virtuales. En cuanto al almacenamiento se ha elegido el que ofrecen por defecto, tipo EBS y de propósito general SSD con un tamaño de 10GB.



The screenshot shows the 'Data nodes storage' configuration page in the AWS console. It includes three configuration fields: 'Data nodes storage type' set to 'EBS', 'EBS volume type*' set to 'General Purpose (SSD)', and 'EBS storage size per node*' set to '10'. A note at the bottom states 'Total cluster size will be 10 GiB (EBS volume size x Instance count)'.

Ilustración 20 Configuración AWS: Tipo de almacenamiento

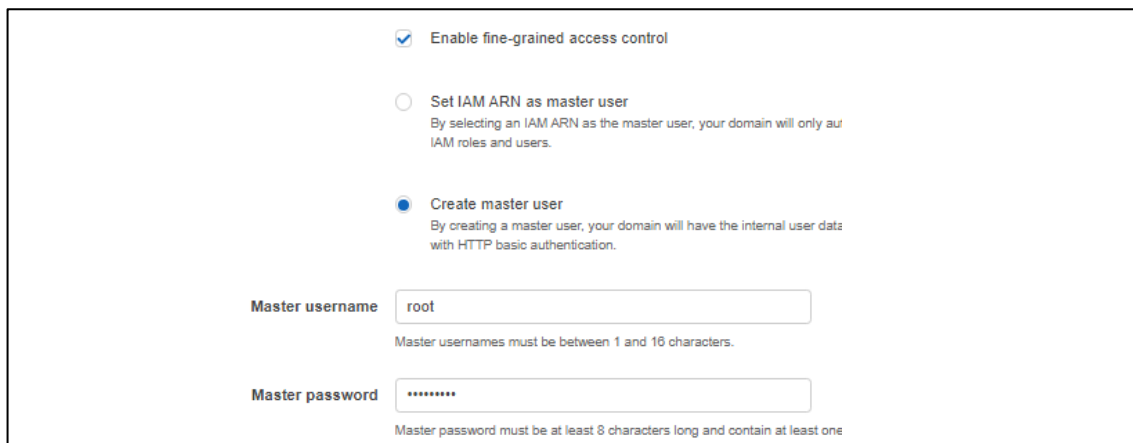
La configuración de un nodo máster dedicado se ha descartado para este proyecto, pero si en un futuro se llevara el desarrollo a producción sería necesario configurar un nodo máster y unos nodos redundantes en base a la demanda esperada.



The screenshot shows the 'Dedicated master nodes' configuration page in the AWS console. It includes a title 'Dedicated master nodes', a descriptive paragraph, and three configuration options: 'Dedicated master nodes' (disabled), 'Instance type' (set to 'r5.large.search'), and 'Number of master nodes' (dropdown menu).

Ilustración 21 Configuración AWS: Nodos máster

En cuanto a la configuración del control de acceso se ha elegido la opción más básica por HTTP con el uso de un usuario máster. Se ha descartado la opción de acceso con IAM ARN para simplificar el acceso. En un desarrollo en producción es muy recomendable utilizar esta última opción porque obtenemos un plus de seguridad en nuestro sistema.



The screenshot shows the 'Access control' configuration page in the AWS console. It features three radio button options: 'Enable fine-grained access control' (checked), 'Set IAM ARN as master user', and 'Create master user'. Below the options are input fields for 'Master username' (containing 'root') and 'Master password' (masked with dots), each with a descriptive note.

Ilustración 22 Configuración AWS: Control de acceso

Una vez todo configurado obtenemos la url del servidor **Anexo 3** que más tarde se utilizará en el script de volcado de datos.

Domain status	Active
Version	Elasticsearch 7.10
Endpoint	https://search-portfolio1-phob4bgj373gac7rqs3kpac7nu.eu-west-3.es.amazonaws.com
Custom endpoint	-
Domain ARN	arn:aws:es:eu-west-3:957524063643:domain/portfolio1
Kibana	https://search-portfolio1-phob4bgj373gac7rqs3kpac7nu.eu-west-3.es.amazonaws.com/_plugin/kibana/
Custom Kibana	-
Availability zones	1
Instance type (data)	t3.small.search
Number of nodes	1
Data nodes storage type	EBS
EBS volume type	General Purpose (SSD)
EBS volume size	10 GiB

Ilustración 23 Configuración AWS: Datos de acceso al nodo

3.3.2. Almacenamiento con Elasticsearch

Como ya se ha comentado, una de las mayores ventajas de Elasticsearch es que cuenta con una API REST integrada que nos permite administrar y consultar los datos indexados. Mediante *queries* podemos obtener documentos indexados, filtrar información, añadir nuevos documentos o eliminarlos.

En todo este proyecto su implementación esta muy presente empezando con el guardado de los documentos que se han obtenido en las fases de scraping, generación de informes, hasta la lectura de documentos para generar el dataset o cumplimentar el portfolio con las predicciones.

Para comunicar la aplicación **Anexo 1** con el servidor se ha utilizado una librería cliente de Elasticsearch oficial llamada `elasticsearch-py` []. Para implementarla en la aplicación, se ha creado un *wrapper* llamado **elastic.py** donde se han añadido las funcionalidades básicas para realizar tanto escrituras como lecturas en el servidor.

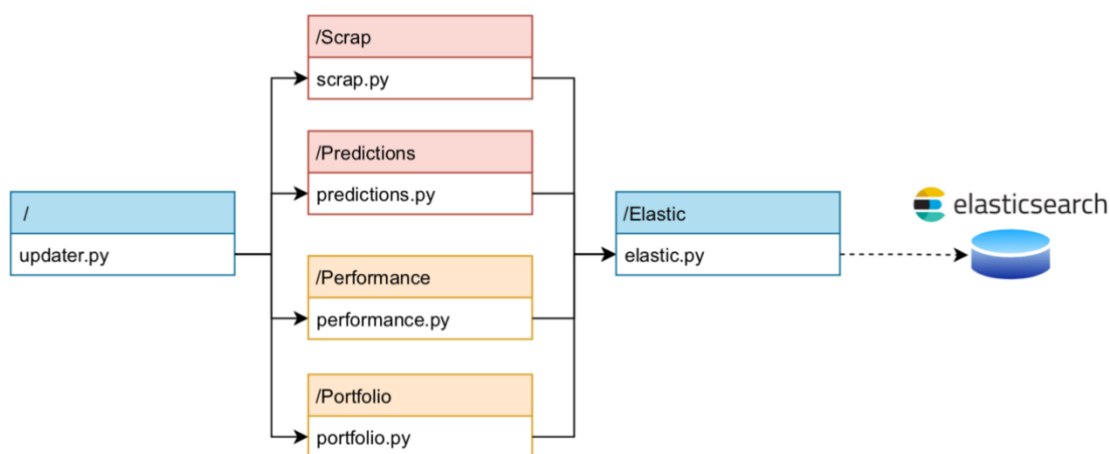


Ilustración 24 Arquitectura del script updater

Para poder conectarnos, únicamente es necesario configurar la url que nos han proporcionado en AWS y configurar en Elastic Stack un usuario y password que tenga permisos de escritura y lectura.

Para mantener al día la información alojada en el servidor se ha creado el script **updater.py** que llama a las distintas funciones que se han comentado en los apartados anteriores. Mediante comandos bash podemos decidir la operación a realizar:

Comando	Descripción
<code>python3 update.py -day</code>	Ejecuta a las funciones <code>scrap.py</code> y <code>predictions.py</code>
<code>python4 update.py -month</code>	Ejecuta a las funciones <code>performance.py</code> y <code>portfolio.py</code>

Tabla 12 Comandos del script updater

Es muy importante poder realizar las operaciones con comandos bash ya que permiten decidir cual es la operación que queremos realizar en todo momento. En nuestro servidor podemos fijar el día o a la hora que queremos que se realice la operación y con los comandos podemos seleccionar la operación.

Las actualizaciones se han dividido en mensuales y diarias. Las primeras se realizan para mantener las predicciones al día. Las compañías publican de forma trimestral sus informes financieros pero a menudo pueden ocurrir cambios importantes que se pueden publicar en cualquier momento siendo necesaria una actualización mensual.

Las segundas son necesarias para mostrar en el dashboard todas las operaciones que se realicen a lo largo del tiempo. Además ayudan a tomar decisiones de compra o venta basándose en las predicciones que se hacen de forma mensual y el precio actual. Finalmente, se mantiene al día el rendimiento de la cartera.

ÍNDICES DE ELASTIC

La base de datos de Elasticsearch almacena las colecciones de datos en índices, los índices permiten tener diferenciados y clasificados los datos. En la siguiente tabla se muestran los índices que se han generado para el almacenaje de los documentos.

Index	Descripción	Uso
portfolio	Se almacenan los datos de las posiciones actuales de DEGIRO	Dashboard
performance	Se almacena el rendimiento de la cartera junto con el índice del SP500	Dashboard
predictions	Se almacenan las predicciones realizadas por el algoritmo de machine learning	Dashboard
cid	Se almacenan información general sobre las empresas que se extraen con el script de scraping.	Creación dataset
financials	Se almacenan el estado financiero de las empresas que se extraen con el script de scraping.	Creación dataset
price	Se almacena el precio de las empresas que se extraen con el script de scraping.	Creación dataset

Tabla 13 Índices de Elasticsearch

ESQUEMAS DE ELASTIC

Además de los índices también es importante establecer el esquema de datos, cuando se almacenan los datos en Elastic, el sistema es capaz de identificar el tipo de cada variable de forma automática pero a veces el resultado no es el esperado. Por esta razón se ha creado un esquema para los índices portfolio, performance y predictions que se encuentran definidos en la carpeta /Data/* en su respectiva carpeta. A continuación se muestra el esquema de predictions.

```
{
  "mappings": {
    "properties": {
      "id": { "type": "long" },
      "company": { "type": "keyword" },
      "currPrice": { "type": "float" },
      "prediction": { "type": "float" },
      "profit": { "type": "float" }
    }
  }
}
```

Ilustración 25 Esquema del índice predictions

Como se puede observar los string son tratados como tipo “keyword” es decir como un listado de elementos únicos, para que posteriormente en la visualización se puedan utilizar.

3.4. Algoritmo de aprendizaje automático

Con el dataset que previamente se ha construido con el script `dataset_creator.py` se ha realizado un estudio en un Notebook con JupyterLab para posteriormente desarrollar un modelo de aprendizaje automático que permita predecir el precio de las acciones.

El estudio **Anexo 2** está centrado en las 500 empresas con mayor capitalización del mercado bursátil americano del SP500, por lo que el modelo sólo será capaz de predecir acciones americanas de alta capitalización. El comportamiento entre mercados puede ser distinto, por lo que en futuras líneas de trabajo se podría desarrollar otros modelos que se centren en el mercado Europeo o en empresas de baja capitalización (small caps).

El dataset está formado por los datos financieros de 189 empresas que van del 2015 al 2021. La variable objetivo es "Next year price", se trata del precio a un año vista que se ha obtenido de TIKR. Los doce ratios elegidos para construir el dataset son los más relevantes desde el punto de vista financiero, la mayoría se utilizan para calcular el valor intrínseco de las acciones por lo que estos deberían aportar mayor poder predictivo.

En la siguiente figura se muestra un pequeño extracto del dataset.

	Year	CID	Currency	Exchange	Exchange Name	Industry	Next year price	Company Name	Revenues	Depreciation & Amortization	...	Interest Expense	Income Tax Expense	Weighted Average Diluted Shares Outstanding	Capital Expenditure	Cash And Equivalents	Current Portion of Long-Term Debt	Long-Term Debt	Goodwill	Total Equity	Cash Flow per Share
0	2015.0	25307.0	USD	NYSE	New York Stock Exchange	Specialty Retail	789.79	AutoZone, Inc.	10187.340	269.919	...	-152.044	-642.371	32.206	-480.579	175.309	NaN	4624.876	391.887	-1701.390	34.298
1	2016.0	25307.0	USD	NYSE	New York Stock Exchange	Specialty Retail	711.37	AutoZone, Inc.	10635.676	297.397	...	-150.052	-671.707	30.488	-488.791	189.734	NaN	4924.119	391.887	-1787.538	38.217
2	2017.0	25307.0	USD	NYSE	New York Stock Exchange	Specialty Retail	838.34	AutoZone, Inc.	10888.676	323.051	...	-158.082	-644.620	29.065	-553.832	293.270	NaN	5081.238	391.887	-1428.377	35.764
3	2018.0	25307.0	USD	NYSE	New York Stock Exchange	Specialty Retail	1191.31	AutoZone, Inc.	11221.077	345.084	...	-180.163	-298.793	27.424	-521.788	217.824	NaN	5005.930	302.645	-1520.355	57.787
4	2019.0	25307.0	USD	NYSE	New York Stock Exchange	Specialty Retail	1185.44	AutoZone, Inc.	11863.743	365.757	...	-192.200	-414.112	25.498	-496.050	176.300	NaN	5206.344	302.645	-1713.851	65.387

5 rows x 21 columns

Ilustración 26 Dataset con los datos financieros

3.4.1. Limpieza de datos

Para empezar a analizar los datos es imprescindible realizar una limpieza de datos. Para empezar se han añadido ceros en aquellos ratios que están vacíos y realmente se trata de que su valor sea cero. Esto es debido a que la página TIKR no asigna ningún valor cuando realmente es cero o prácticamente inexistente. Un ejemplo de ello es la deuda a corto y largo plazo, existen algunas empresas del SP500 que no tienen deuda por lo que se debe asignar cero.

Para continuar se han eliminado las filas con valores faltantes considerando que rellenarlos con la media desvirtuará la predicción y reducirá el poder predictivo del algoritmo. También, se han descartado los datos del año 2020 por ser un año atípico debido a la pandemia y los del año actual ya que estos se utilizarán más adelante para predecir el precio del año 2022.

```
[8]: df['Current Portion of Long-Term Debt'] = df['Current Portion of Long-Term Debt'].fillna(0)
df['Long-Term Debt'] = df['Long-Term Debt'].fillna(0)
df['Goodwill'] = df['Goodwill'].fillna(0)
df['Capital Expenditure'] = df['Capital Expenditure'].fillna(0)
```

Ilustración 27 Limpieza de datos

3.4.2. Correlación y estudio de las variables

Una vez realizada la corrección de los datos se ha analizado la correlación entre variables dependientes e independiente. Para ello se ha aplicado la matriz de correlación de Pearson que se muestra a continuación.

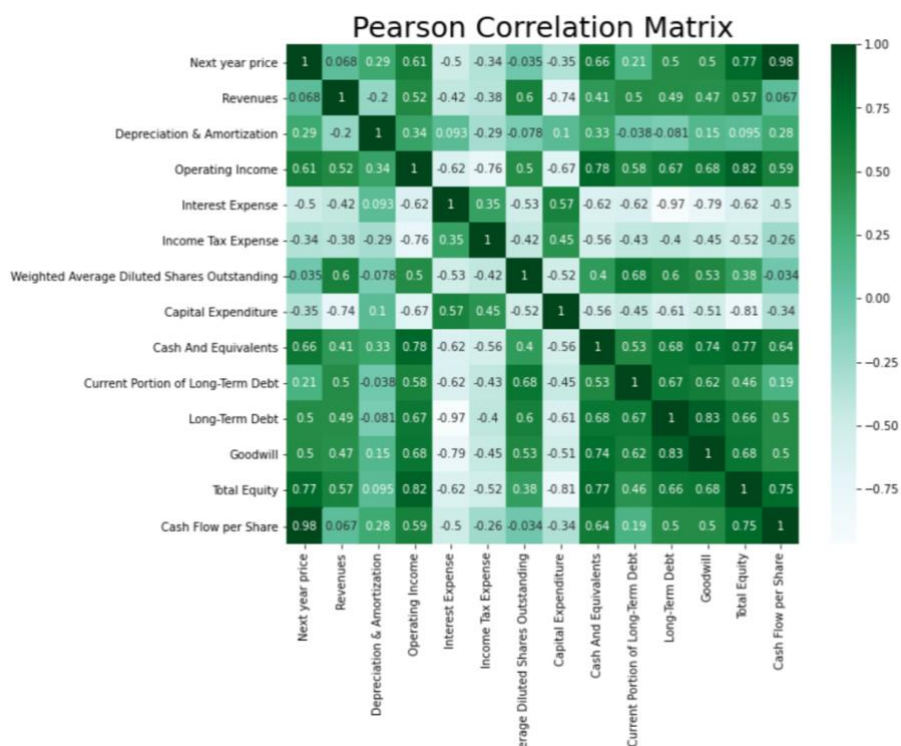


Ilustración 28 Matriz de correlación Pearson

Como se puede observar en la matriz hay muchas variables que no aportan poder predictivo por lo que se pueden descartar, poniendo el foco en las diez variables con mayor correlación con el precio.

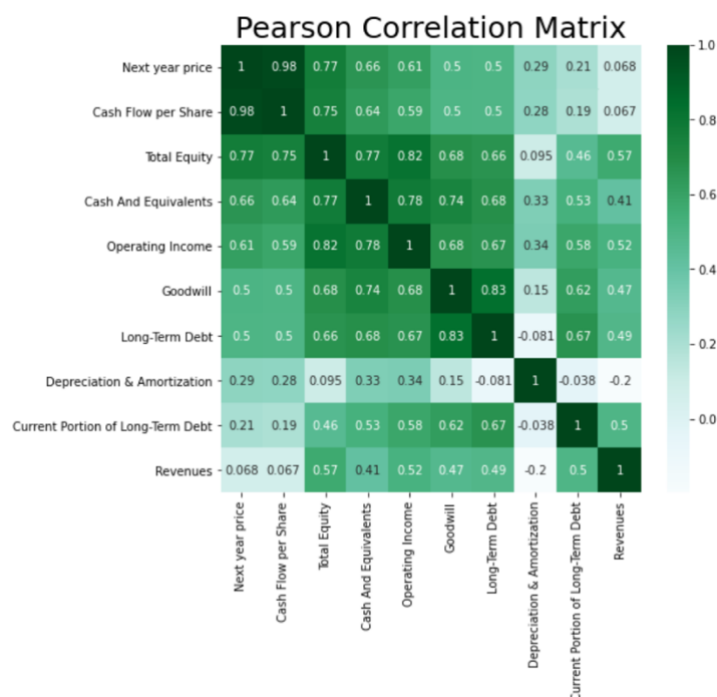


Ilustración 29 Las diez variables con mayor correlación

Las variables que están más correlacionadas con el precio son el Cash flow per share con un 0.98, el Total equity con un 0.77 y Cash and Equivalents con 0.66. El problema que existe es que entre ellas se encuentran fuertemente correlacionadas entre sí por lo que sólo podemos seleccionar una.

En este caso se elige el Cash Flow per share ya que es el que más correlación existe con el precio y el Current Portion of Long-Term Debt ya que es el que menos correlación existe con el Cash flow y aporta cierto poder de predicción.

En cierto modo, esta correlación de 0,98 nos indica que el método del descuento de flujos de caja para determinar el precio de una acción, que se ha explicado en el estado del arte del proyecto es muy acertado ya que existe una fuerte correlación con el precio y el free cash flow.

Una vez elegidas las variables que se utilizarán en los modelos de aprendizaje automático se realiza la separación de los datos de test y entrenamiento. Se ha elegido un 20% de los datos para test de forma aleatoria y el resto para el entrenamiento del algoritmo.

3.4.3. Selección de los algoritmos

Aunque al principio del proyecto se quería plantear como un problema de clasificación se ha tomado la decisión de considerarlo como un problema de regresión ya que no tenemos un punto de referencia que nos permita reconocer si deberíamos mantener, comprar o vender la acción.

Además, para un inversor aporta más la información de conocer el precio objetivo esperado ya que cuando el precio de la acción se encuentre cerca del objetivo puede considerar vender o mantener según otros factores que no considerará el algoritmo como el análisis cualitativo del sector, o país en el que se desarrolla.

Los algoritmos de regresión elegidos son la regresión lineal, Support Vector Machine Regression, Random Forest Regressor y una red neuronal.

A continuación se muestra las estadísticas obtenidas con los algoritmos de aprendizaje una vez entrenados los modelos y testados con los datos de test. Obtenemos el error medio absoluto MAE, error cuadrático medio MSE, raíz del error cuadrático medio RMSE y el coeficiente de determinación R2 Square.

Modelo	MAE	MSE	RMSE	R2 Square
Lineal regression	1118.582913	1.694141e+08	13015.918800	0.797076
Support Vector Machine - linear	1060.546511	1.993522e+08	14119.212294	0.761216
Support Vector Machine - poly	4234.361037	3.074259e+09	55446.003656	-2.682349
Random Forest	519.710926	2.289911e+07	4785.301250	0.972571
Artificial Neural Network	2901.319111	8.430279e+08	29034.942454	-0.009779

Tabla 14 Estadísticas de los modelos entrenados

El algoritmo que mejor determina los datos es el **Random forest** y es el que además el error es más pequeño. Observamos un error de una magnitud elevada debido a que hay empresas que cotizan a un precio muy alto como por ejemplo Berkshire Hathaway a 427.701,00 USD o Alphabet a 2.827,36 USD.

Con la red neuronal se han obtenido mejores resultados que los que se muestran en los datos posteriormente pero siguen siendo inferiores a los obtenidos con el Random Forest, por lo que se ha decidido descartarlo.

3.4.4. Aplicación del algoritmo al script

Notebook

Una vez elegido el modelo que mejor predice el precio de la acción, se ha utilizado la librería `joblib` que permite guardar un modelo entrenado en un fichero, para posteriormente ser utilizado con la misma librería con las variables a predecir. Mediante `compress=3` se comprime el fichero resultante.

```
import joblib
joblib.dump(rf_reg, "random_forest.joblib", compress=3)
```

Script con Elastic

Para predecir los precios de las acciones se ha creado un script llamado **predictions.py**. El script primero obtiene los datos financieros que previamente se han scrapeado y guardado en el servidor de Elastic. Obtiene las variables más actualizadas “Cash Flow per Share” y “Current Portion of Long-Term Debt” del año en curso así como también el precio actual de la acción.

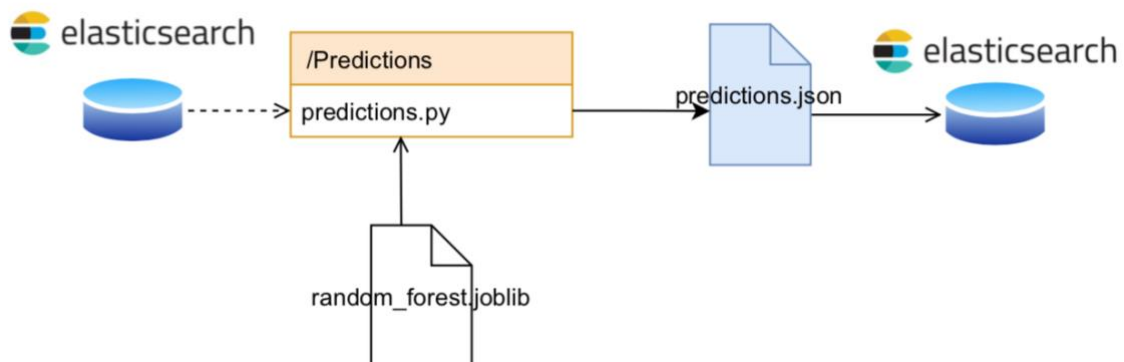


Ilustración 30 Arquitectura del script predictions

Con la librería `joblib` se importa el modelo Random Forest previamente entrenado y lo ejecuta con las ratios obtenidos. El algoritmo retorna los precios predichos y calcula el beneficio que obtendríamos, considerando el precio actual.

```
loaded_rf = joblib.load("random_forest.joblib")
test = cleandf[['Current Portion of Long-Term Debt', 'Cash Flow per Share']]
pred = loaded_rf.predict(test)
```

La predicción se obtiene en formato array por lo que es necesario posteriormente añadirlo en un fichero JSON junto con los datos que se utilizarán en el servidor de Elastic, como el nombre de la empresa, precio actual, beneficio porcentual que puede ser negativo en el caso de que el precio determinado pueda caer.

La estructura que se ha aplicado al fichero JSON es la siguiente:

Nombre	Descripción
id	Id interno de la empresa
company	Nombre de la compañía
currPrice	Precio actual
prediction	Precio predicho
profit	Beneficio porcentual esperado

3.5. Visualización de los datos con Kibana

Kibana es un fronted de código abierto que se encuentra integrado en Elastic Stack, cuenta con la capacidad de visualización y búsqueda de datos indexados en bases de datos Elasticsearch, además de ofrecer funcionalidades para monitorizar y gestionar un clúster.

El entorno nos brinda con una serie de herramientas muy potentes que nos ayudan a analizar nuestros datos mediante gráficos de barra, circulares, tablas e histogramas. Con la combinación de estos elementos podemos crear dashboards y analizar a tiempo real grandes cantidades de datos.

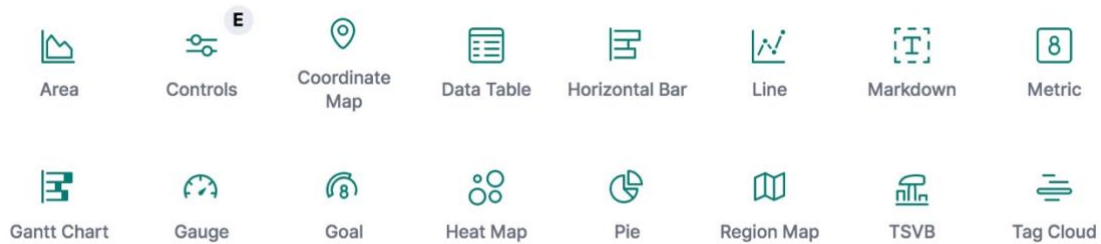


Ilustración 31 Kibana: Tipos de visualizaciones disponibles

3.5.1. Diseño del dashboard

Antes de realizar el dashboard con Kibana **Anexo 3** se ha realizado un pequeño esquema con distintos recortes para decidir los gráficos que van a formar parte del dashboard, su distribución y el diseño sin tener en cuenta las limitaciones que existen en el sistema. A continuación se muestra el esquema realizado:

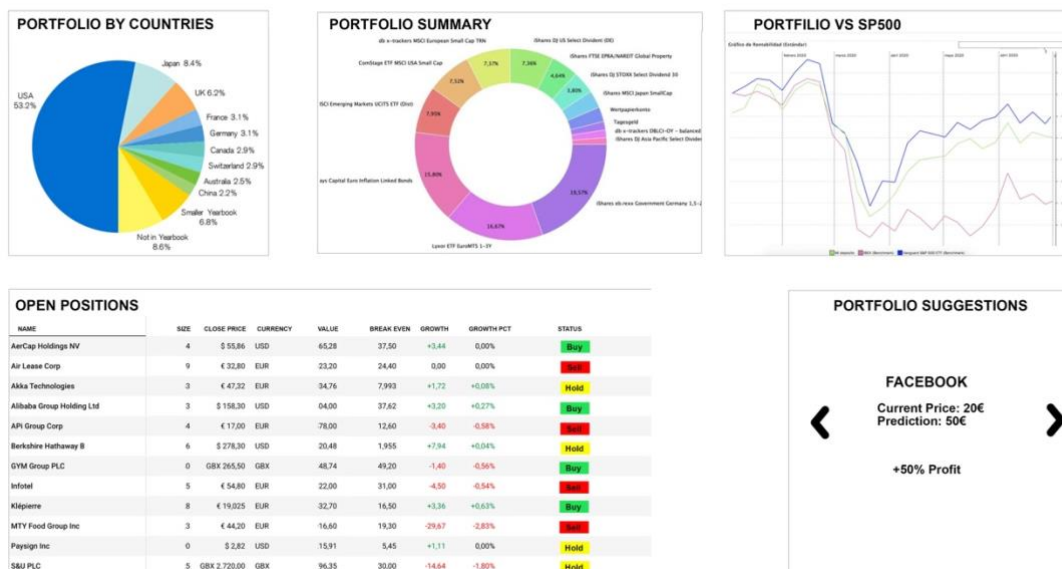


Ilustración 32 Distribución del dashboard

Portfolio by countries

El primer gráfico muestra el porcentaje de acciones por países, lo que nos ayudará a verificar si nuestra cartera esta bien diversificada o si debemos compensarla.

Portfolio summary

El segundo gráfico se trata de un resumen del número de acciones que tenemos de cada empresa, nos permiten saber si debemos ajustar el número de acciones para estar mejor diversificados.

Portfolio vs SP500

El tercer gráfico es el rendimiento acumulado de nuestra cartera en contraposición al índice de referencia SP500. Sabemos que estamos realizando buenas inversiones cuando estamos por encima del índice, cuanto mayor sea el margen mayor será el beneficio.

Open positions

Es una tabla resumen con todas las posiciones actuales, muestra el precio actual de la acción, número de acciones, valor de mercado x el número de acciones, precio medio de la inversión, crecimiento absoluto y porcentual y finalmente la acción recomendada por el algoritmo de machine learning que puede ser comprar, vender o mantener.

Portfolio suggestions

Muestra sugerencias de inversión, se trata de las empresas que el algoritmo de machine learning analiza y las ordena de mayor a menor beneficio. Muestra el valor actual del momento en que se ejecuto la predicción y el precio predicho.

3.5.2. Configuración del dashboard

Kibana requiere patrones de índice para poder acceder a los datos de Elasticsearch. Un patrón de índice permite seleccionar los datos que se utilizarán para la visualización y definir las propiedades de los campos. Cuando configuramos un patrón podemos seleccionar un índice haciendo coincidir una o varias fuentes.

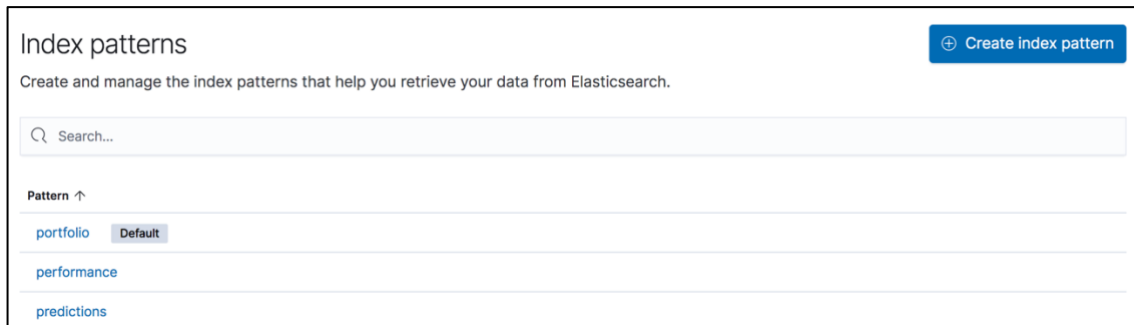


Ilustración 33 Kibana: Index Patterns configurados

Para construir el dashboard con los elementos visuales se ha configurado un patrón de índice para portfolio, predictions y performance. Previamente se ha configurado su esquema al realizar el volcado en el servidor, por lo que los *data types* de todos los campos se encuentran configurados.











Name	Type	Format	Searchable	Aggregatable	Excluded
<u>_id</u>	string		•	•	
<u>_index</u>	string		•	•	
<u>_score</u>	number				
<u>_source</u>	<u>_source</u>				
<u>_type</u>	string		•	•	
company	string		•	•	
currPrice	number		•	•	
id	number		•	•	
prediction	number		•	•	
profit	number		•	•	

Ilustración 34 Kibana: Campos de los índices

Cuando configuramos un índice podemos hacer una previsualización de los campos que lo forman además de sus propiedades. El tipo de datos se configura en los esquemas de tablas aunque si no se le añade un esquema el sistema le asigna de forma automática un tipo. Los tipos son muy importantes ya que según su configuración nos permiten realizar búsquedas o agregar datos.

Además existen un tipo de variable especial que se indican con el guion bajo, estas variables las utiliza el sistema para mejorar el rendimiento y hacer más rápidas las búsquedas. En este proyecto se ha utilizado el _id para asignarle el Company ID para agilizar las búsquedas.

3.5.3. Visualización de los datos

Una vez configurados los Index Patterns, Kibana puede acceder a los datos y solo queda añadir los elementos visuales que formarán nuestro dashboard. A continuación se detalla la configuración de cada uno de los elementos.

PORTFOLIO POR PAÍSES

La primera visualización se ha configurado con un gráfico del tipo Pie. Accediendo al índice portfolio y al campo "country" se ha configurado la métrica count. Kibana cuenta el número de veces que se repite cada elemento de country y muestra su valor porcentual.

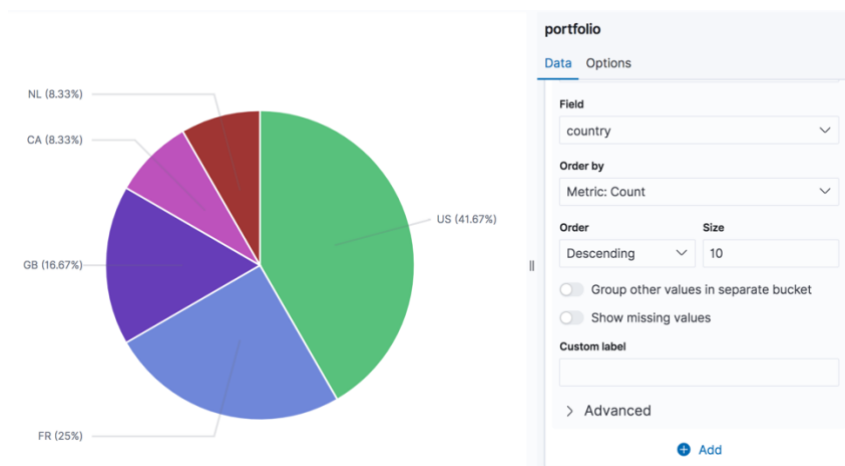


Ilustración 35 Kibana: Portfolio por países

RESUMEN DEL PORTFOLIO

La segunda visualización se ha utilizado también un gráfico del tipo Pie. Desde el índice de portfolio accede a los campos value y name. Con el campo value que contiene el peso de cada compañía según el número de acciones y el precio actual de mercado realiza una suma y lo divide por el total para obtener el peso de cada compañía en nuestro portfolio.

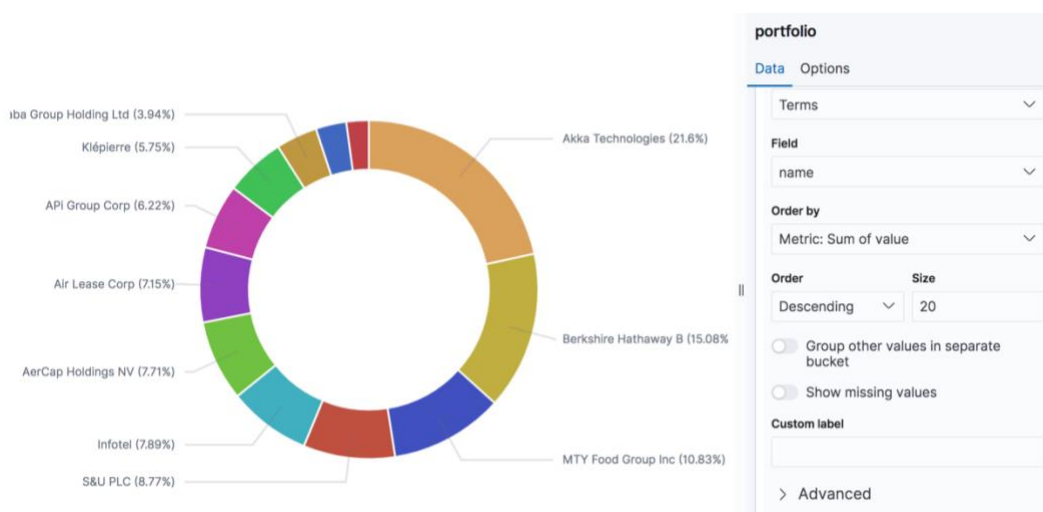


Ilustración 36 Kibana: Resumen del portfolio

PORTFOLIO VS SP500

Para la siguiente visualización se ha utilizado un gráfico de líneas. En el que se ha trazado dos líneas de tendencia. En la primera se ha configurado el rendimiento acumulado de la cartera, y en la segunda el rendimiento del SP500. Los datos los obtiene del índice performance, las fechas previamente se han configurado en el Index Pattern como referencia de tiempo para el eje X, y se han configurado los campos “p” y “s” para el eje Y.

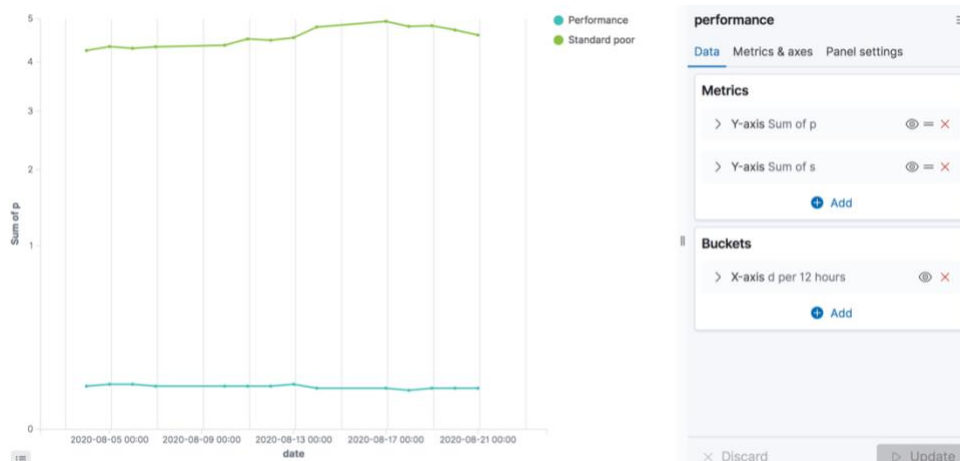


Ilustración 37 Kibana: Portfolio vs SP500

SUGERENCIAS DE INVERSIÓN

Para esta visualización se ha utilizado un tabla ya que es la forma más sencilla de poder filtrar y ordenar posteriormente los datos. Los datos se obtienen del índice predictions, que proviene del algoritmo de machine learning. Los campos company, current price, predictions se han configurado para ser mostrados de forma ascendente y el de profit de forma descendente para ver en la primera página, las compañías con las que obtendremos mayor rentabilidad.

Company	Current Price	Prediction	Profit	Count
Centrale d'Achat Française pour l'Outre-Mer Société Anonyme	159	258	0.623	1
Deceuninck NV	125	112	-0.104	1
Elecnor, S.A.	109	245	1.248	1
Esprinet S.p.A.	118	174	0.475	1
Etteplan Oyj	76	91	0.197	1
Fontaine Pajot SA	212	189	-0.108	1

Export: [Raw](#) [Formatted](#)

1 2 3 4 »

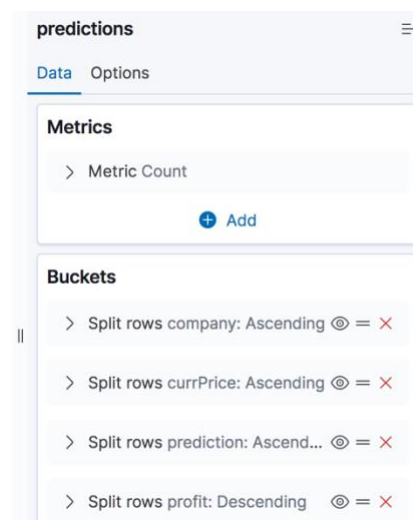


Ilustración 38 Kibana: Sugerencias de inversión

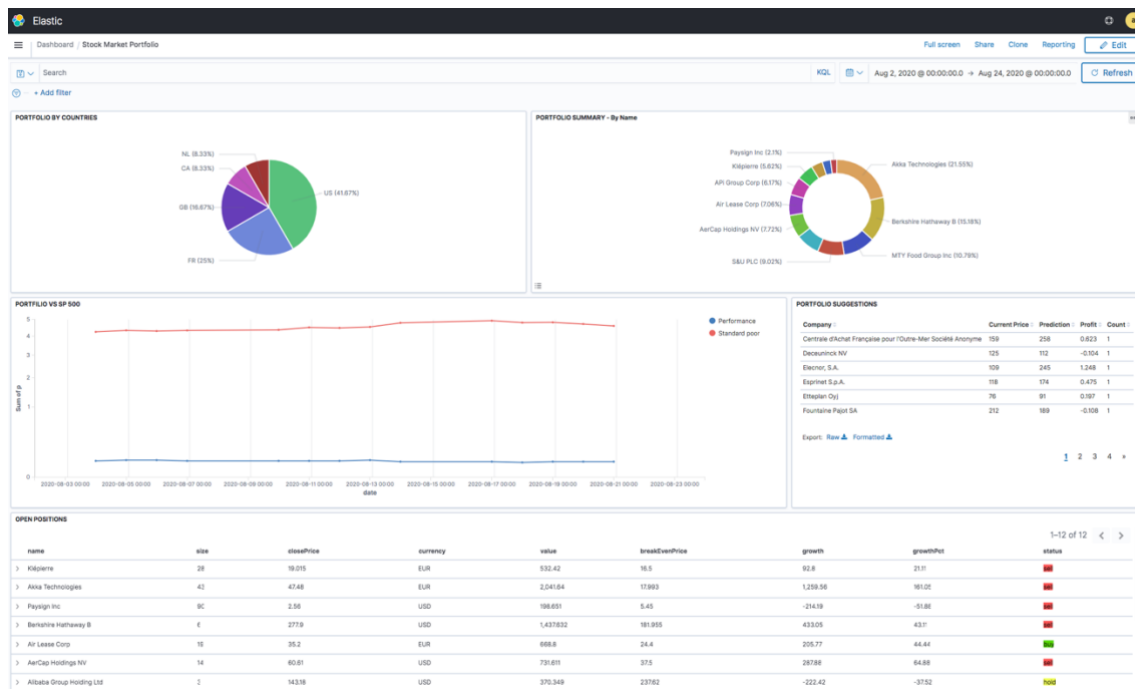
POSICIONES ABIERTAS

Para el último gráfico se ha configurado una tabla de búsqueda donde se ha añadido el índice porfolio con todos sus campos. Además se ha configurado un semáforo para el campo status, dependiendo de la acción que debemos realizar.

name	size	closePrice	currency	value	breakEvenPrice	growth	growthPct	status
Klépierre	28	19.015	EUR	532.42	16.5	92.8	21.11	sell
Akka Technologies	43	47.48	EUR	2,041.64	17.993	1,259.56	161.05	sell
Paysign Inc	90	2.56	USD	198.651	5.45	-214.19	-51.88	sell
Berkshire Hathaway B	6	277.9	USD	1,437.632	181.955	433.05	43.11	sell
Air Lease Corp	19	35.2	EUR	668.8	24.4	205.77	44.44	buy
AerCap Holdings NV	14	60.61	USD	731.611	37.5	287.88	64.88	sell
Alibaba Group Holding Ltd	3	143.18	USD	370.349	237.62	-222.42	-37.52	hold
GYM Group PLC	80	285	GBX	267.9	149.2	206.67	337.53	buy
S&U PLC	25	2,910	GBX	854.813	1,630	426.09	99.76	buy
Infotel	15	50.8	EUR	762	31	310.41	68.74	hold
APi Group Corp	34	17.2	EUR	584.8	12.6	148.51	34.04	hold
MTY Food Group Inc	23	44.43	EUR	1,021.89	19.3	572.27	127.28	sell

Ilustración 39 Kibana: Posiciones abiertas

Una vez configurados todos los elementos del dashboard se puede modificar la distribución, y el tamaño de los gráficos. A continuación se muestra el resultado final obtenido.



Capítulo 4. CASO DE NEGOCIO

4.1. Introducción

En la última década han aparecido numerosas empresas tecnológicas del sector financiero que brindan servicios de planificación financiera automatizada, sin ningún tipo supervisión humana basándose en algoritmos de aprendizaje automático e integrando lo que popularmente se conoce como Robo-Advisors.

Robo-Advisor que proviene de las palabras “Robo” de robot y “Advisor” de asesor, se tratan de gestores automatizado que gestionan el dinero por ti. Seleccionan inversiones, optimizan tu cartera y elaboran un plan según los objetivos de cada uno.

Muchas de las empresas que ofrecen estos servicios no dan muchas pistas de cómo juega la inteligencia artificial en sus funciones, cualquier cosa que hagan para hacer crecer tú dinero o obtener ganancias es prácticamente un secreto para los usuarios.

Aunque la dinámica es siempre la misma, se proporciona al usuario una cartera de acciones diversificada, adaptada a su riesgo. Para ello, en el registro se debe rellenar una encuesta y se asigna un tipo de perfil. La IA es capaz de equilibrar la cartera, hacer sugerencias específicas según la situación de la cartera, estado del mercado, con el objetivo de hacer crecer nuestro dinero.

4.2. Estudio de mercado

A continuación se ha realizado un estudio de mercado con los competidores más importantes en el panorama internacional y nacional. Con especial foco en su modelo de negocio, plataformas y precio.

SAIVERS

Saivers [37][38] es una empresa de asesoramiento financiero con sede en Miami (EEUU), que combinan la inversión a largo plazo con la inteligencia artificial ,para crear carteras sostenibles en el tiempo que generen un rendimiento superior.

Utilizan la inteligencia artificial para identificar empresas que están sufriendo cambios estructurales que alteren antiguas industrias, o creen nuevos mercados sostenibles en el tiempo. Con especial foco en aquellas que tengan un equipo gestor excelente.

Podemos acceder a sus servicios mediante la página de Zonavalue, se trata maketplace de productos de inversión en español. En el encontramos 6 estrategias de inversión que podemos elegir según nuestro perfil.

Zonavalue es un Marketplace español de productos de inversión donde sus socios pueden acceder a 6 carteras construidas por Saivers. Aseguran que en ninguna de sus carteras hay intervención humana, sus portfolios están contruidos por “agentes de inteligencia artificial”.

Cada cartera tiene su rentabilidad publicada de forma trimestral y anual y podemos ver que de media tienen un rentabilidad del 12% siendo “SAIVERS – SNP495” la que mejores resultados, que únicamente invierte en las empresas del SP500.

El servicio sólo esta disponible para los socios del club zonavalue, para usuarios gold su precio es de 250€ anuales y una cuota de entrada 990€.



InvestorAI [39] es una compañía con sede en Inglaterra centrada en el mercado de India, aunque este año van a empezar a ofrecer servicios para clientes de UK.

Se definen como un analista de inversiones personal que ofrece ideas de inversión, impulsadas a través de una inteligencia artificial. Se centran tanto en análisis técnico como en los fundamentales de las empresas para realizar las predicciones.

Su público objetivo son inversores tanto ocasionales como habituales a expertos. Ofrecen tanto alertas predictivas, ideas de inversión, optimización de la cartera y ejecución de las transacciones desde la propia aplicación. También disponen de una API para acceder a sus módulos de IA con los que podemos desarrollar aplicaciones de escritorio.

Tienen un modelo de negocio de suscripción mensual, los usuarios deben pagar £14,99 al mes para tener acceso a la plataforma y ofrecen el primer mes gratis de prueba. Sólo disponible desde dispositivos móvil y iOS.



Waelthfront [40] es de servicios de inversión automatizada con sede en Palo Alto, California. Ofrece servicios de inversión, prestamos hipotecarios y servicios de ahorro para la jubilación.

Cuando nos registramos nos piden contestar un pequeño cuestionario donde evalúan nuestro perfil de riesgo y luego elabora un plan individual que va la inteligencia artificial hace evolucionar según nuestro comportamiento.

Los algoritmos rastrean al usuario y automáticamente muestran recomendaciones relacionadas con la diversificación, minimizar el pago de impuestos y maximizar los retornos. Reequilibran la cartera, diversifican los depósitos, también cuenta con un modo automático donde ellos toman el control de las acciones y se encargan de comprar y vender.

Su modelo de negocio es parecido al de los fondos de inversión, cobran una tarifa anual del 0,25% sobre el promedio ponderado de la cartera. Tienen disponible una plataforma de escritorio y también esta disponible para dispositivos móviles Android y iOS.

Betterment

Betterment [41] es una empresa con sede en Nueva York de asesoramiento financiero que ofrece inversiones automatizadas, inversión guiada y planes de jubilación. Su modelo de negocio se centra en inversiones para minorías, asesores financieros y pequeñas y medianas empresas.

Como en otras empresas del sector piden rellenar un cuestionario para determinar el perfil de inversión y configuran un portfolio diversificado. A diferencia de otros Betterment esta principalmente centrada en ETF's que son híbridos entre acciones y fondos de inversión.

Cuentan con un sistema de rebalanceo de cuenta, reinvierten los beneficios que obtienes, pueden vender automáticamente las acciones para evitar pérdidas a nivel fiscal y además tienen una herramienta de ayuda para evitar pagar impuestos innecesarios.

El modelo es el mismo que Wealthfront y además comparten las mismas tarifas, un 0,25% anual sobre el promedio total de la cartera. Tienen disponible una plataforma de escritorio y también esta disponible para dispositivos móviles Android y iOS.



Moneyfarm [42] es otra empresa centrada en robo advisors con sede en Reino Unido. Su servicios son prácticamente los mismo que Betterment y Wealthfront. Están centrado en inversiones automáticas de ETF's.

Su modelo de negocio esta basado en una tarifa porcentual que depende de la inversión que tengamos. Para una inversión de £10.000 la tarifa que se le aplica es de 0,75% mensual, lo que supone unos £90 anual, una inversión de entre £10.000 – £50.000 se aplica un 0.60%, para una inversión de £50.000 – 100.000 unos 0.50% y más de £100.000 unos 0.35%.

Tienen versión de escritorio y versión para dispositivos móviles,

4.3. Resumen de mercado

En definitiva los sistemas de inversión automática han revolucionado el mercado gracias a la inteligencia artificial. Las funcionalidades más comunes son el reequilibrio de las carteras y sugerencias de inversión.

Una buena parte de ellas están centradas en la inversión de ETF's, aunque también encontramos enfocadas a acciones. Además de ofrecer servicios de inversión automática también ofrecen servicios de ahorro, planes de jubilación basados en inversión y servicios hipotecarios.

Su público objetivo está orientado a pequeños inversores, sobretodo inexpertos en la materia que quieran una pequeña rentabilidad anual. Parte del atractivo de estos servicios son las tarifas anuales más bajas que promedian un 0.25% anual, en comparación con un gestor patrimonial o un fondo de inversión que las comisiones pueden rondar entre 5% y el 10% según la gestión y el éxito.

Uno de los argumentos a favor de los asesores humanos en contra posición con los sistemas automáticos es la gestión emocional del dinero y la tolerancia al riesgo. Los asesores sirven de puente con los clientes para que puedan preguntar y conocer la estrategia que se está siguiendo y entender que está pasando con su dinero, cuando se trata de inversiones en bolsa pueden haber momentos en que los precios caigan y genere ciertas dudas la inversión, el papel del asesor en estos casos es fundamental para explicar la situación y no caer en las trampas emocionales de la inversión.

	Saivers	InvestorAi	Waelthfront	Betterment	Moneyfarm
Año fundación	2020	2017	2008	2008	2011
Sede	Miami, EEUU	London, UK	Palo Alto, EEUU	New York, EEUU	London, UK
Uso de la IA	Carteras diversificadas con distintos perfiles de riesgo	Indicadores y sugerencias de inversión	Perfil de riesgo del cliente, reequilibrar cartera. reducir impuestos	Perfil de riesgo del cliente, reequilibrar cartera. reducir impuestos	Perfil de riesgo del cliente, reequilibrar cartera.
Tipo de inversión	Acciones	Acciones	Acciones	ETF's	ETF's
Rentabilidad	Rentabilidad auditadas	Rentabilidades no auditadas, dan algunas pistas	Poco o nada transparentes	Poco o nada transparentes	Poco o nada transparentes
Dispositivos	Sólo web	iOS, Android	Web, iOS, Android	Web, iOS, Android	Web, iOS, Android
Transacciones automáticas	No	Si	Si	Si	Si
Público objetivo	Inversores experimentados que quieren mejorar rentabilidades	Inversores inexpertos y profesionales	Inversores inexpertos sin experiencia en bolsa	Inversores inexpertos sin experiencia en bolsa	Inversores inexpertos sin experiencia en bolsa

4.4. Plan de negocio

PROPUESTA DE VALOR

El big data junto con la inteligencia artificial esta provocando una disrupción en todo tipo de mercados. Empresas en antiguos sectores que han aplicado esta nueva tecnología han adquirido una ventaja competitiva muy difícil de superar por aquellos competidores que no la poseen.

El mundo financiero es uno de ellos, gracias a las nuevas tecnologías son cada vez más los jóvenes que se interesan por la inversión en bolsa y tienen mayor conocimiento debido en parte por el fácil acceso a la información y la facilidad de aprender.

Este hecho ha impulsado las plataformas de inversión y brokerage online, donde pequeños inversores tratan de obtener un pequeño beneficio anual para protegerse de la inflación o tener unos ahorros para una futura jubilación.

La propuesta de valor es ofrecer herramientas de inversión para inversores tanto experimentados con no tan experimentados y fondos de inversión que quieran mejorar sus rentabilidades, aportándoles ideas de inversión y sugerencias en sus carteras actuales que les ayude a mejorar sus rentabilidades utilizando inteligencia artificial.

EL PROBLEMA

La anticipación es un factor clave en la inversión, son muchas las variables que hacen fluctuar los precios de las acciones, si podemos anticiparnos a los cambios podemos mejorar las rentabilidades. Con la ayuda de un sistema automático que realice un seguimiento de todas las variables el inversor no deberá preocuparse por hacer un seguimiento exhaustivo de las compañías que tiene en su cartera.

MODELO DE NEGOCIO

Los clientes tendrán acceso a una plataforma con 5 carteras de inversión creadas con aprendizaje automático que podrán elegir según sus preferencia y nivel de riesgo. Para ello deberán pagar una tarifa fija mensual a la que se ofrecerá un descuentos para pagos trimestrales o anuales.

Una vez el cliente seleccione la cartera no tendrá acceso a las demás, aunque tendrán la posibilidad de cambiar pagando una tarifa adicional. Cada cartera estará formada alrededor de 13 empresas que pueden ser tanto acciones, bonos o ETF's.

Para los usuarios más experimentados podrán crear su propia cartera personalizada donde la IA del sistema sugerirá cambios y nuevas ideas de inversión para mejorar su rendimiento.

La plataforma podrá conectarse con los brokers más populares como Interactive Brokers o Degiro pudiendo ejecutar transacciones de forma automática si el cliente lo desea, además de alarmas y notificaciones cuando se produzcan cambios importantes en las estrategias.

Por una tarifa adicional se ofrecerá acceso a los datos financieros y noticias de las empresas actualizadas de forma periódica con la que los inversores podrán hacer sus propios estudios y tomar sus propias decisiones.

MERCADO OBJETIVO

El mercado objetivo de esta plataforma esta centrada en Europeos con dos tipos de perfil diferenciados, un perfil profesional con dedicación plena al mundo de las finanzas y otro público no tan profesional pero con conocimientos sólidos sobre el tema.

El primer perfil esta centrado en un público profesional como gestores de fondos de inversión o sociedades de inversión (SICAV's) que quieran mejorar el rendimiento de su cartera o reducir su tiempo en buscar nuevas inversiones, o necesiten información actualizada de los datos financieros de las empresas.

El segundo perfil se trata de un público no tan profesional de entre 24 a 39 años que conocen muy bien las nuevas tecnologías y el potencial que ofrecen, que tengan ya conocimientos sobre el mundo financiero, con capacidad de ahorro y un pequeño capital con el que poder empezar a invertir.

PRODUCTO CARACTERISTICAS Y PRECIO

Para empezar se ofrece un paquete Basic por 15€ al mes donde los clientes podrán elegir una cartera de inversión gestionada con Machine Learning que contará con 13 empresas con potencial de crecimiento. Además, la plataforma se podrá conectar con los brokers más importantes que hay ahora mismo a nivel Europeo como Interactive Brokers y Degiro. Las transacciones se ejecutarán de forma automática si el usuario lo desea por lo que no deberá hacer nada.

Basic	Advance	Premium
15€ AL MES	30€ AL MES	47€ AL MES
<ul style="list-style-type: none">- 1 Cartera de inversión ML- 13 Empresas con potencial de crecimiento- Conexión con los Brokers más populares.- Ejecución automática transacciones	<ul style="list-style-type: none">- Lo mejor del Basic- 5 carteras de inversión ML- Más de 50 empresas con potencial de crecimiento.- Acceso a datos financieros y noticias de las compañías	<ul style="list-style-type: none">- Todo lo del advance- Carteras personalizadas- Sugerencias de inversión- Recomendaciones para mejorar el rendimiento

El paquete Advance por 30€ al mes ofrece lo mejor del paquete Basic con la posibilidad de tener acceso a cinco carteras distintas que en todo momento podrá consultar. Además de tener acceso a las últimas noticias del mercado y a los datos financieros de los dos mercados principales más importantes el Europeo y el Americano.

Para terminar el paquete Premium por 47 al mes para inversores más exigente que requieran un control exhaustivo de sus propias posiciones y quieran obtener recomendaciones y sugerencias que realizará el algoritmo de aprendizaje automático.

Capítulo 5. PRESUPUESTO

A continuación se detalla el presupuesto para desarrollar el proyecto, se incluyen las horas de trabajo para 4 desarrolladores, material, viajes, formación y servicios informáticos y gastos en publicidad.

El coste total para este proyecto es de 223,100€ para desarrollar una primera fase beta funcional donde los primeros clientes responderán a cuestionarios para mejorar la plataforma y solucionar los errores que puedan haber en la programación.

FECHA DE INICIO: 01 Agosto de 2021

DURACIÓN DEL PROYECTO: 12 meses

CONCEPTO	NÚM. UNIDADES	COSTE UNIDAD	TOTAL
Gastos de Personal			
Costes directos por personal (1 año) - Full Time	4	45,000.00 €	180,000.00 €
Material fungible			
Material de oficina (papelería, bolígrafos, toners...)	4	150.00 €	600.00 €
Material inventariable			
Ordenadores (Portátiles)	4	1,200.00 €	4,800.00 €
Monitores	4	300.00 €	1,200.00 €
Impresoras	1	200.00 €	200.00 €
Servidor para base de datos	1	30€/mes	300.00 €
Viajes y dietas			
Reuniones con clientes	-	3,000.00 €	3,000.00 €
Acceso a ferias	-	2,000.00 €	2,000.00 €
Gastos en publicaciones			
Publicidad redes sociales	-	20,000.00 €	20,000.00 €
Publicación de artículos en la prensa	-	5,000.00 €	5,000.00 €
Gastos en formación			
Contratación de cursos de formación (para personal directo)	3	2,000.00 €	6,000.00 €
Gastos en servicios informáticos			
Instancia AWS ES t3.small.search	3	25/mes	900.00 €
Costes totales			223,100.00 €

Tabla 15 Presupuesto del proyecto

Capítulo 6. PLANIFICACIÓN

A continuación se detalla la planificación del proyecto para 52 semanas, teniendo en cuenta que trabajarán 4 personas en el proyecto y se dividirá en cinco fases que se desarrollarán en paralelo.

En la fase de desarrollo del algoritmo se trabajará en la predicción del precio de las acciones, estudio de los datos, evaluación de los algoritmos y integración del modelo en el backend.

La fase de desarrollo de la plataforma web y de la aplicación móvil se trabajará en un diseño sencillo y fácil de entender par el usuario y se continuará con la integración del diseño en el backed.

En la fase de backend se configurará la base de datos y se realizará el desarrollo del API REST para conectar el frontend de las distintas plataformas con la base de datos.

La última fase se realizará la corrección de los errores que se hayan encontrado en las fases de prueba y testeo.

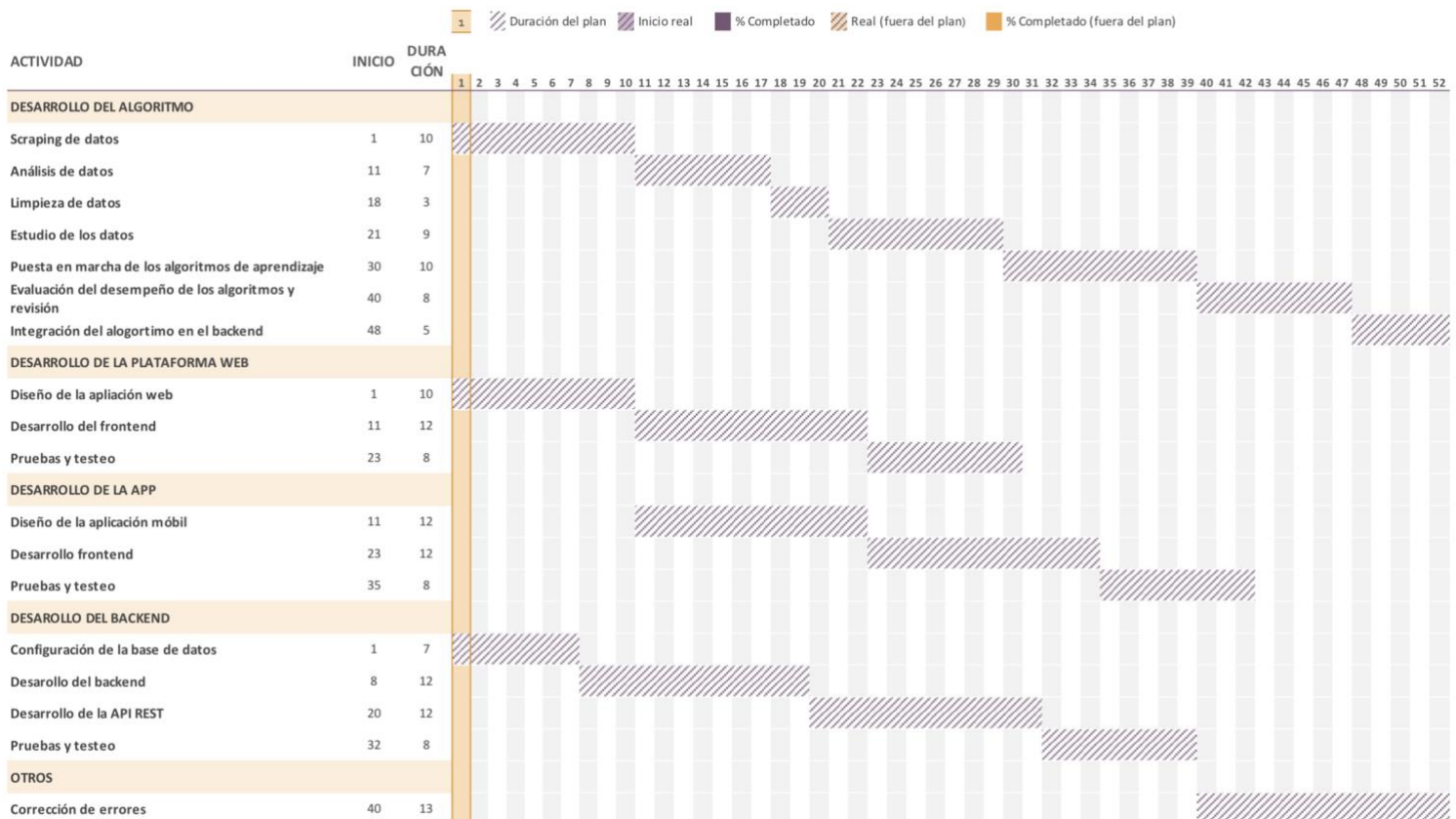


Ilustración 40 Planificación del proyecto

Capítulo 7. CONCLUSIONES Y FUTURAS LÍNEAS DE TRABAJO

En este proyecto se ha puesto en práctica muchos de los conceptos que se han visto a lo largo del máster. Para empezar se ha trabajado con una base de datos NO SQL como Elasticsearch. Aunque en el curso se han visto otras bases de datos, ha sido muy interesante poder ampliar conocimientos y aprender desde cero nuevas herramientas.

Además, se ha trabajado con algoritmos de aprendizaje automático los cuales se han aplicado en un caso real y práctico surgiendo problemas con los datos, teniendo que realizar tests de prueba y error hasta dar con el que mejor encaja con los requisitos.

También se ha trabajado conceptos en la visualización de datos que se han visto en el curso, como los visual encodings, saturación de elementos, simplicidad en las visualizaciones, fases de desarrollo de una visualización, realización de un sketch o la definición de los elementos que se quieren mostrar en el dashboard.

De las asignaturas de creación y estrategia de empresa, se han aplicado los conceptos aprendidos en el capítulo del caso de negocio, donde se ha elaborado un plan de empresa, un estudio de mercado para conocer los posibles competidores, definición del modelo de negocio y del público objetivo.

Ha supuesto un gran reto poder desarrollar este sistema de toma de decisiones desde cero y lograr que esté perfectamente cohesionado y funcionando con las bases de datos y los scripts. Aunque ha sido muy satisfactorio ya que el sistema es capaz de ayudar a un inversor a mejorar sus rendimientos con sólo un simple dashboard.

Algunos de los puntos a mejorar en este proyecto sería por ejemplo la fiabilidad de los algoritmos de aprendizaje automático. Se debería centrar en una tipología concreta de acciones para mejorar las predicciones y reducir el universo de posibilidades.

Otro punto es la ampliación de los mercados bursátiles, actualmente el sistema de aprendizaje automático está únicamente centrado en acciones del mercado americano, principalmente porque es uno de los mercados con mayor liquidez del mundo y ofrece muchas oportunidades de inversión. Pero se podría aplicar a mercados emergentes como por ejemplo la India.

Algunas de las mejoras que se presentan en el plan de negocios son la posibilidad de poder tener distintas carteras según el perfil del inversor, actualmente sólo está centrada en las empresas más importantes del SP500. Por lo que se podría ofrecer otros tipos de inversión que vayan más allá de las acciones como los bonos o los ETF's.

Aprovechando los datos financieros que tenemos ya alojados en el servidor se podría ofrecer esta información a los clientes para que puedan hacer sus propios análisis y estar al día de las últimas actualizaciones.

Finalmente se debería incluir un sistema de registros de usuarios, con el sistema actual se pueden añadir nuevos usuarios con distintos permisos de acceso pero se debería incluir una pasarela de registro y un sistema de pago por tarjeta.

ANEXOS

ANEXO 1: Scripts de actualización de Elasticsearch, scraping y obtención de datos de DEGIRO:
<https://github.com/msanc/PortfolioAI>

ANEXO 2 : Jupyter Notebook con el desarrollo del algoritmo de aprendizaje automático
<https://github.com/msanc/NotebookAI/blob/master/TFM.ipynb>

ANEXO 3: Acceso al dashboard de Elasticsearch – Global - Stock Market Portfolio
https://search-portfolio1-phob4bgj373gac7rqs3kpc7nu.eu-west-3.es.amazonaws.com/plugin/kibana/app/kibana_overview#/

User: viewer

Pass: Europea2021%

BIBLIOGRAFÍA

1. **Patel, J., Shah, S., Thakkar, P., & Kotecha, K.** (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, 42(1), 259-268.
2. **Lv, D., Yuan, S., Li, M., & Xiang, Y.** (2019). An empirical study of machine learning algorithms for stock daily trading strategy. *Mathematical Problems in Engineering*, 2019.
3. **Zhong, X., & Enke, D.** (2019). Predicting the daily return direction of the stock market using hybrid machine learning algorithms. *Financial Innovation*, 5(1), 1-20.
4. **F. Norrestad**, Total number of accounts of online broker DEGIRO [Consulta: 5 de Agosto de 2021]. Disponible en: [Consulta: 5 de Agosto de 2021]. Disponible en: <https://www.statista.com/statistics/963078/degiro-accounts/>
5. **Kibana**. *Elastic Stack Kibana* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.elastic.co/es/kibana/>
6. **Tikr Terminal**. *Financial company data* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://app.tikr.com/>
7. **Raisin**. *Value investing* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.raisin.es/inversion/value-investing-la-importancia-del-largo-plazo/>
8. **Paco Lodeiro**. *El margen de seguridad* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.academiadeinversion.com/margen-de-seguridad-value-investing-definicion-calculo-importancia/>
9. **Viktor Tachev**. *Análisis Fundamental – Una Guía Completa de Básico a Avanzado*. [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://blog.earn2trade.com/es/analisis-fundamental/>
10. **José Francisco López**. *Análisis Fundamental* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://economipedia.com/definiciones/analisis-fundamental.html>
11. **Francisco J.** *Métodos de valoración de empresas*[Consulta: 20 de Setiembre de 2021]. Disponible en: <https://franciscobarrionuevo.eu/files/2015/03/ENSAYO-M%C3%89TODOS-DE-VALORACI%C3%93N-DE-EMPRESAS.pdf>
12. **Paco Lodeiro**. *La fórmula de Benjamin Graham para calcular el valor intrínseco de una acción* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.academiadeinversion.com/margen-de-seguridad-value-investing-definicion-calculo-importancia/>
13. **Madhuri Thakur**. *Intrinsic Value Formula* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.wallstreetmojo.com/intrinsic-value-formula/>
14. **Manish Choudhary**. *Discounted Cash Flow (DCF)* [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.academiadeinversion.com/margen-de-seguridad-value-investing-definicion-calculo-importancia/>

15. **Afroz Chakure.** *Random Forest Regression.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f>
16. **Jose Martinez Heras.** *Random Forest (Bosque Aleatorio): combinando árboles.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://www.iartificial.net/random-forest-bosque-aleatorio/>
17. **TIBCO.** *What is a Random Forest?.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://www.tibco.com/reference-center/what-is-a-random-forest>
18. **Salleh, M. N. M., Talpur, N., & Talpur, K. H. (2018).** *A modified neuro-fuzzy system using metaheuristic approaches for data classification.* Artificial Intelligence–Emerging Trends and Applications. London, United Kingdom: IntechOpen, 29-45. [Consulta: 5 de Agosto de 2021]. Disponible en: <https://www.intechopen.com/chapters/60337>
19. **Neel Shelar.** *Explain ANFIS architecture with a neat diagram.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://neelshelar.com/explain-anfis-architecture-with-a-neat-diagram/>
20. **Salleh, Mohd Najib Mohd; Talpur, Noureen; Hussain, Kashif.** *Adaptive neuro-fuzzy inference system: Overview, strengths, limitations, and solutions.* En *International Conference on Data Mining and Big Data.* Springer, Cham, 2017. p. 527-535. [Consulta: 5 de Agosto de 2021]. Disponible en: https://www.researchgate.net/publication/318154996_Adaptive_Neuro-Fuzzy_Inference_System_Overview_Strengths_Limitations_and_Solutions
21. **Scikit-Learn.** *Support vector machines.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://scikit-learn.org/stable/modules/svm.html>
22. **Madhu Sanjeevi.** *Support Vector machine with Math.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://medium.com/deep-math-machine-learning-ai/chapter-3-support-vector-machine-with-math-47d6193c82be>
23. **Saedsayad.** *Support Vector Machine.* [Consulta: 5 de Agosto de 2021]. Disponible en: https://www.saedsayad.com/support_vector_machine.htm
24. **John, McGonagle; José, Alonso; Saruque, Mollick.** *Feedforward neural network.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://brilliant.org/wiki/feedforward-neural-networks/>
25. **T&M Equipments.** *Feedback Control vs Feedforward Control-Difference between Feedback Control and Feedforward Control.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://www.test-and-measurement-world.com/Terminology/Difference-between-Feedback-Control-and-Feedforward-Control.html>
26. **Reza Bagheri.** *An Introduction to Deep Feedforward Neural Networks.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://towardsdatascience.com/an-introduction-to-deep-feedforward-neural-networks-1af281e306cd>

27. **Kaivan Kamali.** *Feedforward neural networks.* [Consulta: 5 de Agosto de 2021]. Disponible en: <https://training.galaxyproject.org/training-material/topics/statistics/tutorials/FNN/tutorial.html>
28. **Kenchannavar, Anita H.;** RODD, S. F. Fundamental Analysis of Financial Parameters and Ranking using Artificial Neural Network. *Bonfring International Journal of Software Engineering and Soft Computing*, 2016, vol. 6, no Special Issue Special Issue on Advances in Computer Science and Engineering and Workshop on Big Data Analytics Editors: Dr. SB Kulkarni, Dr. UP Kulkarni, Dr. SM Joshi and JV Vadavi, p. 134-139
29. **HUANG, Yuxuan.** Machine learning for stock prediction based on fundamental analysis. 2019.
30. **Huang, Yuxuan; CAPRETZ, Luiz Fernando;** HO, Danny. Neural network models for stock selection based on fundamental analysis. En 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE). IEEE, 2019. p. 1-4
31. **Degiro.** Bróker Degiro. [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://www.degiro.es/>
32. **Elasticsearch.** ¿Qué es Elasticsearch? [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.elastic.co/es/what-is/elasticsearch>
33. **Refinitiv.** Tecnología financiera y datos [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://www.refinitiv.com/es>
34. **Bram Tom.** Unofficial DeGIRO Python API [Consulta: 20 de Setiembre de 2021]. Disponible en: <https://github.com/bramton/degiro>
35. **Python Elasticsearch.** Python Elasticsearch Client [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://elasticsearch-py.readthedocs.io/en/v7.15.0/>
36. **Amazon Web Services.** ¿Qué es OpenSearch? [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://aws.amazon.com/es/opensearch-service/the-elk-stack/what-is-opensearch/>
37. **Zona Value.** Mundo de la inteligencia artificial en la inversión [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://zonavalue.com/estrategias-ia>
38. **Saivers.** Saivers investing with IA [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://saivers.com/>
39. **InvestorAI.** Investing app with IA [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://investorai.co.uk/>
40. **Wealthfront.** Make Wealth Your Own [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://www.wealthfront.com/>
41. **Betterment.** Investing & Saving App [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://www.betterment.com/>
42. **Moneyfarm.** Investing that gives you more [Consulta: 29 de Setiembre de 2021]. Disponible en: <https://www.moneyfarm.com>