Universidad Popular Autónoma del Estado de Puebla

Vicerrectoría Académica

Decanato de Ingenierías

Aplicación de metodologías de aprendizaje automático para hacer predicciones en el mercado de valores

Tesis para obtener el Grado de Maestro en Ciencia de Datos e Inteligencia de Negocios

Presentada(o) por:
Adrian Barradas Barradas

Directora de tesis

Dra. Rosa María Cantón Croda

H. Puebla de Zaragoza, México. enero 2022

Universidad Popular Autónoma del Estado de Puebla

Vicerrectoría Académica

Decanato de Ingenierías

Maestría en Ciencia de Datos e Inteligencia de Negocios

Se aprueba la Tesis/Proyecto Práctico:

"Aplicación de metodologías de aprendizaje automático para hacer predicciones en el mercado de valores"

Nombre del alumno:
Adrian Barradas Barradas

Comité Asesor

Dra. Rosa María Cantón Croda
Directora de Tesis/Proyecto Práctico

Dr. Damián Emilio Gibaja Romero
Asesor

Dr. Vittorio Zanella Palacios
Asesor

H. Puebla de Zaragoza, México enero 2022

Resumen

El deseo por predecir el comportamiento del mercado de valores se remonta a principios del siglo XVIII; desde entonces se han planteado diversas propuestas para cumplir con ese objetivo, y más recientemente estos esfuerzos se han hecho utilizando técnicas de aprendizaje automático. Este proyecto presenta una propuesta de una herramienta predictiva para predecir el porcentaje de variación en el precio diario de las cincuenta acciones más representativas del índice bursátil S&P500, así como la determinación de las condiciones en el mercado de valores a partir de noticias financieras, datos transaccionales e indicadores económicos. El desarrollo se lleva a cabo siguiendo una metodología CRISP-DM en la que se aplican técnicas de minería y almacenamiento de datos, y algoritmos de aprendizaje automático: redes neuronales entrenadas por retropropagación, y agrupamiento por k medias. Los resultados principales muestran que la integración de noticias financieras contribuye significativamente a mejorar la precisión de los modelos.

Palabras clave: aprendizaje automático, redes neuronales, agrupamiento por k medias, mercado de valores; predicción

Abstract

The desire to predict stock market behavior dates back to the early seventeenth century; since then, several proposals have been put forward to meet that goal, and more recently these efforts have been made using machine learning techniques. This project presents a proposal for a predictive tool to predict the percentage change in the daily price of the fifty most representative stocks of the S&P500 stock index, as well as the determination of stock market conditions from financial news, transactional data and economic indicators. The development is carried out following a CRISP-DM methodology in which data mining and warehousing techniques, and machine learning algorithms are applied: neural networks trained by backpropagation, and k-means clustering. The main results show that the integration of financial news contributes significantly to improve the accuracy of the models.

Keywords: machine learning, neural networks, k-means clustering, stock market, prediction.

Índice general

Índice de Tablas

Índice de Figuras

Índice de Ecuaciones

Índice de Gráficas

Introducción

Hoy en día, en el mercado de valores de Estados Unidos se realizan alrededor de 15 mil millones de transacciones diarias (Cboe Global Markets, 2021). En el mercado de valores los inversionistas tienen acceso a la compra y venta de instrumentos financieros, entre los que se encuentran las acciones de un gran número de empresas. La inversión en acciones representa un riesgo para el inversionista en el sentido de que el dinero invertido puede perder valor; sin embargo, también existe la posibilidad que incremente su valor. En este contexto, el inversionista debe tomar las decisiones adecuadas que lo lleven a incrementar el valor de su patrimonio bursátil; ante este hecho, el factor psicológico juega un papel importante, pues, de acuerdo con las finanzas conductuales, los inversionistas no siempre se comportan de forma racional y, en su lugar, actúan basados en sus emociones (López-Cabarcos, Pérez-Pico, Vázquez-Rodríguez, & López-Pérez, 2019) y en recomendaciones o creencias basadas en experiencias previas (De Long, Shleifer, Summers, & Waldmann, 1990). Tomando como base la hipótesis del mercado eficiente (EMH), que plantea que el mercado se comporta de forma racional (Ritter, 2003) y que responde a los eventos que acontecen en su entorno; el presente trabajo tiene como objetivo generar un modelo predictivo que integra datos de noticias financieras, índices económicos y datos transaccionales de las cincuenta acciones más representativas del índice bursátil S&P500 dentro del mercado de EE. UU para estimar la variación de su precio futuro. Para realizar la estimación se proponen la aplicación de algoritmos de aprendizaje automático: *redes neuronales* y *agrupamiento por k medias*.

Este proyecto se desarrolla utilizando tres herramientas informáticas: *Python (Jupyter Notebooks)*, *MySQL* y *Matlab*; y siguiendo una metodología de cuatro fases basada en la metodología de minería de datos CRISP-DM. La primera fase corresponde a la comprensión del problema; en ella se establece la necesidad de la solución, y se contextualiza el problema. La segunda fase es la más extensa ya que en ella se realiza la preparación de los datos. De acuerdo con (Press, 2016), la preparación de los datos representa alrededor del 80% del trabajo en un proyecto de ciencia de datos. Las tareas que se ejecutan en esta fase son: recopilación, comprensión y transformación de los datos para ser cargados a una base de datos multidimensional que integra y relaciona los datos en una tabla de hechos. Los datos se recolectan de forma diaria, por lo que el volumen de datos es elevado tomando en cuenta que el estudio considera un periodo de seis años

que va del 01 de enero 2014 al 31 de diciembre de 2020. La preparación de los datos se lleva a cabo utilizando diversas herramientas disponibles en el repositorio público de librerías de *Python* (*Python Package Index)*, entre las que se encuentran librerías para la recolección de datos mediante *Web Scrapping*, y para el procesamiento de texto de lenguaje natural y análisis de sentimientos para clasificar las noticias financieras según su grado de positividad o negatividad. Por último, se establece una conexión con *MySQL* para cargar los datos en la base de datos multidimensional.

En la tercera fase, se lleva a cabo el modelado de los datos; en ella se hace una revisión de dos modelos de redes neuronales en *MatLab*: un modelo para predecir el porcentaje de variación diaria en el precio de las acciones, y otro para clasificar el sentido de la variación en el precio de las acciones. En *Python* también se revisa un modelo de agrupamiento por *k medias* para categorizar las condiciones en el mercado bursátil. En todos los casos se toma como base para el análisis la tabla de hechos del modelo multidimensional. Los modelos revisados, en conjunto son de utilidad para la toma de decisiones pues tanto los dos primeros modelos que estiman la variación en el precio de las acciones, como el modelo de agrupamiento por *k medias* proporcionan información sobre el comportamiento esperado en el mercado de valores basado en datos históricos. La implementación del proyecto sucede en la cuarta y última fase. En ella se desarrolla una aplicación en *Python (Jupyter Notebooks)* desde la que el usuario es capaz de generar un reporte, que, a partir de una fecha y código bursátil correspondiente a una acción, muestra una predicción para la variación en el precio, las condiciones en el mercado, y una recomendación sobre si es un momento adecuado para invertir. Par el correcto funcionamiento de la aplicación se establece comunicación entre *Python, MySQL* y *Matlab* para que los procesos correspondientes se puedan ejecutar sin necesidad de intervención por parte del usuario.

A partir de los modelos estudiados en este proyecto se determinó que la integración de noticias financieras es significativa para obtener predicciones más precisas. El modelo de red neuronal que no integra las noticias financieras tiene un menor rendimiento en comparación con el modelo que sí las integra. Por otra parte, el modelo de agrupamiento por *k medias* además de considerar el sentimiento de las noticias; considera el nivel de popularidad de las acciones en los medios de comunicación; es decir, qué tanto se habla de la acción independientemente de la polaridad del

sentimiento. Este enfoque ayuda a identificar las condiciones en el mercado y tiene como finalidad mostrar al inversionista una estimación del panorama general en el mercado.

En el presente trabajo se aplican técnicas y metodologías propias de la ciencia de datos para generar una herramienta que proporciona al usuario inversionista información sobre el mercado de valores con la finalidad de que pueda tomar mejores decisiones. La propuesta planteada en este proyecto, según mi mejor conocimiento, es una de las primeras que integra técnicas diversas como minería de datos, gestión de bases de datos, transformación y limpieza de datos, y aprendizaje automático supervisado y no supervisado en el área bursátil para la obtención de conocimiento entorno a ella. Cabe destacar que este trabajo aporta valor debido a que considera diversos factores que afectan el comportamiento del mercado de valores: indicadores técnicos del análisis bursátil, sentimiento del mercado basado en noticias financieras e índices económicos.

.

1 Propósito y Organización

1.1 Planteamiento del Problema

El inversionista desea que su patrimonio incremente con el paso del tiempo, para ello busca minimizar pérdidas y aumentar las ganancias en cada inversión; razón por la cual, la toma de decisiones es fundamental. De acuerdo con Chandra (2008) la toma de decisiones en el mercado de valores usualmente está influenciada por factores conductuales como el miedo, la codicia, y la heurística; y por factores sociales como el comportamiento gregario (Ngoc, 2014). Las finanzas conductuales estudian los efectos psicológicos sobre el comportamiento de los inversionistas y sostienen que los inversionistas actuán de forma irracional. Por otro lado, la hipótesis del mercado eficiente afirma que a pesar de la irracionalidad de los inversionistas, el mercado se comporta de forma racional (Ritter, 2003); en otras palabras, los instrumentos bursátiles reflejan toda la información existente, y sus precios se ajustan a ella; teniendo esto en cuenta, el inversionista debe mantener un portafolio diversificado en el que considere la correlación entre las acciones elegidas.

Un portafolio óptimo no se crea únicamente mediante la combinación de instrumentos financieros cuya relación riesgo-rendimiento los caractericen como ideales; también es necesario considerar la relación entre ellos (Reilly & Brown, 2011). La diversificación reduce dicha relación y el riesgo de pérdidas; sin embargo, factores conductuales como la aversión a las pérdidas, el exceso de confianza, y la percepción del riesgo de los inversionistas, así como las tendencias en el mercado, influyen en las decisiones que toma el inversionista (Goetzmann & Kumar, 2008; Alquraan, Alqisie, & Shorafa, 2016). De acuerdo con Subash (2012), los factores anteriores conducen al inversionista a tomar posturas de manera precipitada y a tener portafolios poco diversificados; en este sentido, el factor psicológico resulta tener un efecto significativo durante la elección de los instrumentos bursátiles. Con el objetivo de que el inversionista sea capaz de actuar de forma racional y en consecuencia minimizar las pérdidas, es importante que las desiciones se tomen en mayor medida basadas en datos. En relación con la afirmación de la hipotesis del mercado eficiente, sobre que los precios de los instrumentos financieros reflejan toda la información disponible en sus precios; y a partir de la afirmación Renault (2020), acerca de la relevancia de la cantidad de datos en la predicción en el mercado de valores, es de interés integrar la mayor cantidad

de datos posible entorno a ellos, y determinar si una mayor cantidad de datos contribuye a mejorar la toma de decisiones (Ang, Goetzmann, & Schaefer, 2011).

1.1 Justificación de la investigación

Esta investigación tiene como beneficio la predicción del comportamiento de un grupo específico de acciones que se transaccionan en el mercado de valores basada en factores macroeconómicos y factores inherentes de las empresas que emiten dichos instrumentos, representados a través de noticias financieras. Asumiendo que la subjetividad en la toma de decisiones disminuye al basarlas mayormente en datos, este proyecto ayuda a mejorar la gestión de los activos financieros; y, en consecuencia, de forma directa o indirecta, ayuda a mejorar la calidad de vida de las personas y de su entorno; a la vez que reduce problemas derivados del estrés financiero (OBS Business School, s. f.). Actualmente existen empresas e instituciones que gestionan activos financieros para personas físicas y jurídicas, por lo que los conocimientos generados pueden ser de utilidad y puestos en práctica para incrementar su eficiencia y rendimientos; sin embargo, una de las principales implicaciones de este proyecto es a nivel personal ya que facilitar el acceso a una herramienta financiera evita la necesidad de recurrir a asesoría con altos costos.

La propuesta planteada contribuye a mejorar el entendimiento de las variables que afectan el precio de los instrumentos financieros. No se ha encontrado evidencia de estudios previos que integren noticias y factores macroeconómicos representados por tipos de cambio, índices de volatilidad, índices de mercados emergentes y desarrollados, o materias primas para generar modelos predictivos utilizando aprendizaje automático que permitan predecir la dirección del precio de acciones en el mercado de valores, maximizar los rendimientos y minimizar el riesgo. Algunos estudios han aplicado técnicas estadísticas para hacer predicciones en el mercado de valores (Elbahloul, 2019); sin embargo, a diferencia del presente proyecto, se limitan a predecir un solo instrumento financiero. En este sentido, esta investigación aporta un antecedente para la aplicación de metodologías de aprendizaje automático, minería de textos y análisis de sentimientos, en la toma de decisiones respecto a un grupo de acciones; y se espera que los resultados sean referencia para estudios posteriores que busquen entender el comportamiento de los instrumentos financieros que se transaccionan en mercado bursátil.

1.2 Objetivos de la investigación

1.2.1 Objetivo general

Generar un modelo predictivo del comportamiento de las cincuenta acciones más representativas del índice S&P 500 dentro del mercado de valores de Estados Unidos (EE. UU.) mediante la aplicación de algoritmos de aprendizaje automático con base en una estrategia de análisis combinado integrando noticias financieras.

1.2.2 Objetivos específicos

Generar una base de datos del sentimiento de las noticias financieras utilizando librerías y algoritmos existentes y de uso libre para Python para representar los factores económicos fundamentales de las empresas correspondientes.

Integrar indicadores económicos (tipo de cambio, índices económicos y materias primas) utilizando un base de datos de los valores transaccionales diarios históricos disponibles en fuentes de datos abiertas, para relacionarlos a nivel global y mejorar el modelo predictivo.

Utilizar los datos transaccionales de las cincuenta acciones más representativas del índice S&P 500 utilizando una base de datos generada a partir de valores históricos, para calcular indicadores técnicos que se puedan utilizar para el aprendizaje automático.

1.3 Preguntas de investigación

De acuerdo con la hipótesis del mercado eficiente, ¿cuál es el impacto de la integración de factores como las noticias financieras y los indicadores económicos, en la generación de modelos de aprendizaje automático en la toma de decisiones en portafolios de inversión?

¿Considerar el sentimiento del mercado basado en las noticias públicas contribuye a mejorar la predicción?

¿Tomar en cuenta la correlación entre activos ayuda a mejorar la eficiencia de un modelo predictivo?

¿Retroalimentar el modelo con los datos de las acciones que conforman el portafolio, ofrece mejores resultados?

1.4 Alcances y limitaciones del proyecto

El presente proyecto se limita a estudiar la relación que existe entre las variables previamente propuestas desde el punto de vista de la ciencia de datos y el aprendizaje automático. Los datos se obtienen de dos fuentes: *Google News* para los datos que representan los factores económicos; y *Yahoo Finance* para los datos transaccionales de las acciones a estudiar; en ambos casos el horizonte temporal es de 2014 a 2020. El análisis de sentimientos de las noticias económicas entorno a los instrumentos financieros se realiza utilizando una librería de código abierto de Python para el procesamiento del lenguaje natural. con el único propósito de estudiar su impacto en el modelo predictivo (Bird, Klein, & Loper, 2009). Por último, respecto a este último, se hace una propuesta utilizando algoritmos de aprendizaje automático que considera la correlación entre diferentes instrumentos financieros, para sugerir un conjunto de acciones que ofrezcan al inversionista opciones de inversión basadas en datos.

Ya que el objetivo del presente trabajo es la aplicación del conocimiento, no se desarrollan algoritmos para la recolección de datos y aprendizaje automático, o software para su aplicación práctica; sin embargo, se plantea una propuesta de integración cuya usabilidad de pueda evaluar y que permita obtener mejores resultados en comparación con estudios previos. Por otra parte, la evaluación del modelo se limita a datos históricos dentro del periodo anteriormente planteado. Una evaluación con datos en tiempo real permanece como una propuesta para trabajos futuros.

1.5 Organización del proyecto

En el capítulo 2 de este proyecto se presenta una revisión de la literatura en relación con el tema de estudio; también se plantea la metodología utilizada para el desarrollo del proyecto, y se definen las herramientas y algoritmos para el procesamiento y modelado de los datos. En el capítulo 3, se realiza la recopilación y exploración de datos, y se revisan dos modelos de redes neuronales: regresión y clasificación, y un modelo de *agrupamiento por k medias*. En el capítulo 4 se hace un análisis y discusión de los resultados. Por último, en el capítulo 5 se presenta la conclusión del proyecto

2 Marco conceptual/teórico

2.1 Invertir en el mercado de valores

El mercado bursátil forma parte del mercado de capitales dentro del sistema financiero, y es el lugar en el que se intercambian instrumentos financieros a corto y largo plazo; dichos instrumentos son activos intangibles de los que se espera se obtengan beneficios a futuro en forma de efectivo, un tipo de este instrumento son los valores, que incluyen los bonos y las acciones (Darškuvienė, 2010). Estas últimas permiten a las empresas incrementar su capital, y a los inversionistas convertirse en propietarios de una fracción de ellas, obteniendo el derecho de recibir la parte proporcional de los beneficios que obtengan las empresas (Najeb M. H., 2013; Kingdom of Saudi Arabia, 2018). Sin embargo, el intercambio de acciones puede representar un riesgo para los inversionistas pues se exponen a la volatilidad (Bhagat, 2019), es decir, a la inestabilidad de los precios en el mercado, lo que puede resultar en pérdidas económicas.

De acuerdo con Bhagat, (2019) cuando una inversión representa mayor riesgo, existe la posibilidad de generar ganancias más altas. Ante este hecho, el factor psicológico juega un papel importante, pues, de acuerdo con las finanzas conductuales, los inversionistas no siempre se comportan de forma racional y, en su lugar, actúan basados en sus emociones (López-Cabarcos, Pérez-Pico, Vázquez-Rodríguez, & López-Pérez, 2019). Es decir, que las decisiones que toman están influenciadas por *su sentimiento.* El sentimiento del inversionista es una creencia sobre la situación futura de una inversión que no se justifica por las circunstancias reales del entorno (Baker & Wurgler, 2007) y puede ser catalogado como optimista, neutral o pesimista. De acuerdo con J. De Long et al. (1990), los inversionistas usualmente eligen su portafolio de inversión de forma irracional, basando sus decisiones en recomendaciones o creencias, situación que desde un punto de vista psicológico puede afectar el rendimiento de la inversión. Idealmente, el inversionista se informa sobre los instrumentos financieros y el estado del mercado bursátil; y previo a hacer una transacción toma en cuenta principalmente acontecimientos políticos y económicos que puedan tener un efecto significativo en el comportamiento de las acciones.

Estudios previos han demostrado que existe una correlación entre eventos políticos, económicos y los precios de las acciones en el mercado de valores (Önder & Şimga-Muğan, 2006; K, S., & R.,

2013). En este sentido, existen tres factores principales que impactan el mercado accionario: eventos centrados en EE.UU., claridad y transparencia en el mercado, y por último, las noticias macroeconómicas y de política monetaria (Baker, Bloomb, Davis, & Sammon, 2019); los factores anteriores son externos al instrumento financiero sobre el que se desee invertir. Además, cabe mencionar que cuando factores como el producto interno bruto (PIB) y el índice de producción industrial aumentan, también lo hace el precio de las acciones; y de forma contraria, cuando se trata del índice de desempleo o el índice de precios al consumidor (Pražák, 2018). Hoy en día, dado que la economía de los países industrializados es globalizada (Cruz, 2007), los tipos de cambio, es decir el precio de una moneda en relación con otra, son un indicador del estado general de la economía (Kogid, Asid, Lily, Mulok, & Loganathan, 2012). Adicional a lo anterior, Sindhu, Hussain Bukhari, & Hussain, (2017) indican que el 65% de la variación en el precio de una acción depende de factores internos como el flujo de caja, la rentabilidad, crecimiento, capitalización y dividendos; es decir, los fundamentales de la empresa que emite el instrumento.

Es importante considerar que, de forma general las economías industrializadas basadas en el modelo neoclásico de acumulación de capitales se caracterizan por tener un crecimiento sostenido en la producción y el consumo per cápita por largos periodos de tiempo (King, Plosser, & Rebelo, 1988); dicho crecimiento está vinculado con la innovación y el conocimiento, así como la globalización y la apertura al comercio internacional (Jones, 2018). Bajo este supuesto, es posible para los inversionistas obtener beneficios dentro de un mercado equilibrado. Sin embargo, dado que existe incertidumbre respecto a las condiciones económicas, factores sociales y eventos políticos a futuro, hacer predicciones en el mercado de valores es una tarea difícil. (Nti, Adekoya, & Weyori, 2019). Además, la decisión de hacer una inversión es un acto subjetivo que depende de las expectativas de ganancia, el costo del instrumento financiero, y de la posibilidad de financiar dicha inversión; así como de las experiencias pasadas de los inversionistas (Virlics, 2013).

Considerando la diversidad de variables que tienen efecto sobre los instrumentos que se transaccionan en el mercado de valores, es usual que los inversionistas estudien el mercado de valores previo a la toma de una decisión de inversión; lo hacen mediante un análisis que puede ser fundamental, técnico o una combinación de ambos. El primero tiene como objetivo analizar la situación económica de la empresa o institución que emite el instrumento financiero de forma

interna (estados financieros, ventas, ingresos, deuda, etcétera) y de forma externa (circunstancias políticas, económicas, geográficas, etcétera) para determinar si el valor del instrumento corresponde con sus fundamentos. El segundo se basa en el supuesto de que los movimientos en el mercado son patrones que se repiten de forma cíclica (Khadjeh Nassirtoussi, Aghabozorgi, Ying Wah, & Ngo, 2014); y utilizan datos históricos del instrumento financiero (precio venta, precio compra, volumen, para generar modelos matemáticos que representen las tendencias pasadas y presentes, y por lo tanto las futuras. El análisis técnico por sí sólo no es una herramienta adecuada para explicar las tendencias en el mercado de valores (Khadjeh Nassirtoussi, Aghabozorgi, Ying Wah, & Ngo, 2014) porque no ofrece un contexto ante el cuál justificar los patrones y comportamientos de los activos en el mercado de valores; por lo que es común que los inversionistas utilicen ambos enfoques de forma combinada, pues la integración de factores internos y externos ofrecen predicciones con mayor precisión y exactitud (Nti, Adekoya, & Weyori, 2019). De acuerdo con (Petrusheva & Jordanoski, 2016), una estrategia combinada basada tanto en análisis fundamental como técnico puede dar mejores resultados que si se hace de forma separada.

Desde esta perspectiva, el análisis ayuda a determinar qué tan correcto es el precio de las acciones en el mercado respecto a su valor real, y a identificar la tendencia de cambio. Partiendo de ello, se asume que es posible hacer predicciones del comportamiento del precio de las acciones basado en datos entorno a ellas; y que la cantidad y calidad de datos resulta de relevancia, pues según Renault (2020), la cantidad de datos es un factor significativo para la predicción en el mercado de valores, y afirma que un mayor número de datos ayuda a mejorar la precisión de los resultados. En este sentido, la recolección, transformación y procesamiento de los datos, son tareas importantes y que se deben ejecutar siguiendo una metodología adecuada.

2.2 Minería de datos

La minería de datos es un campo de las ciencias de la computación que se enfoca en el filtrado y descubrimiento de nueva información a partir de grandes cantidades de datos (Baker R. S.). Una característica de la minería de datos es la detección de patrones mediante la identificación de anomalías en los datos utilizando principalmente herramientas computacionales; entre ellas el aprendizaje automático, que es fundamental para la extracción de información y conocimiento (Hand, 2015). Las tareas a ejecutar en un proyecto de minería de datos pueden ser complejas, por

lo que es recomendable seguir una serie de pasos que sirvan de guía. La metodología CRISP-DM (Cross Industry Standard Process for Data Mining) fue desarrollada por un consorcio empresarial integrado por Daimler, SPSS y NRC, y sugiere seis fases para el desarrollo de un proyecto, a la vez que proporciona un resumen del ciclo de vida de la minería de datos (Azevedo & Santos, 2012). La metodología se ha vuelto un estándar para el desarrollo de proyectos de minería de datos, pues proporciona un marco para el análisis y resolución de problemas mediante niveles de abstracción que van de lo general a lo específico; además, facilita la gestión del proyecto, lo que hace que los resultados sean fiables y repetibles (Wirth & Hipp, 2000; Huber, Wiemer, Schneider, & Ihlenfeldt, 2019). En este contexto, el presente proyecto se desarrolla basado en la metodología CRISP-DM y sus seis fases que se describen a continuación.

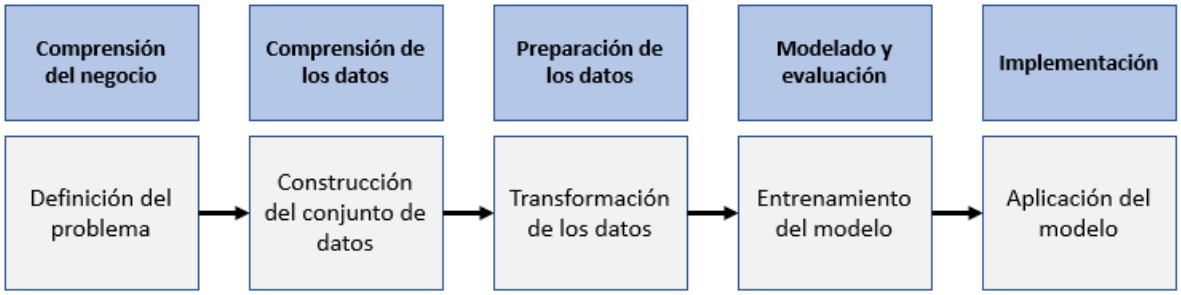


Figura 2.1 Metodología resumida para la minería de datos. Fuente: Elaboración propia con información de (IBM Analytics, 2015)

2.2.1 Comprensión del negocio.

Esta fase se centra en entender los objetivos del proyecto y las necesidades que resuelve, con el propósito de elaborar un plan inicial. Para llevar a cabo esta primera etapa de la metodología, se propone utilizar el método de cuatro pasos *challenge-driven innovation* desarrollado por la empresa *Innocentive* especializada en innovación y resolución de problemas; mismo que se describe a continuación (Spradlin, 2019):

1. Establecer la necesidad de la solución: Para ello, es necesario responder a las siguientes preguntas: ¿Cuál es la necesidad de hacer predicciones en el mercado de valores? ¿Cuál es el resultado esperado? ¿Cuáles son los beneficios y por qué?

2. Justificar la necesidad: En este punto se identifica la razón por la cuál es necesario resolver el problema.
3. Contextualizar el problema: Revisar casos y estudios anteriores para conocer las propuestas que han sido planteadas previamente para evitar invertir recursos en las mismas soluciones. En este punto también es importante definir los alcances y limitaciones, así como los recursos necesarios para el desarrollo.
4. Redactar el planteamiento del problema: Capturar de manera general la información de los tres pasos anteriores; el propósito es tener una visión rápida al problema y que sea inteligible para cualquier persona.

2.2.2 Comprensión de los datos.

En esta fase el problema se expresa bajo el contexto de las técnicas analíticas y aprendizaje automático para definir los requerimientos de los datos, e identificar el modelo a implementar (clasificación, regresión, etc.) (IBM Analytics, 2015). Se inicia con la recopilación de los datos, seguido por su exploración para evaluar su calidad, identificar problemas, y visualizar patrones que ayuden a plantear una hipótesis inicial. En el marco del presente proyecto, esta fase se refiere a la recolección de datos cuantitativos y cualitativos; los primeros corresponden a las transacciones diarias de instrumentos financieros en el mercado bursátil, por ejemplo, precio de compra, precio de venta, volumen de la transacción, y parámetros calculados a partir de dichos valores; mientras que los segundos corresponden a las noticias más relevantes, y están representados por cadenas de texto.

Los datos se obtienen de registros secundarios y existentes; se trata de datos que fueron previamente recolectados, y que, de acuerdo con Drew, (2008) se clasifican como datos de documentos oficiales ya que fueron registrados y almacenados por organizaciones públicas o privadas. Los datos cualitativos y cuantitativos se obtienen mediante un mismo método "*intramethod mixing*", y pueden conservar su valor numérico o textual, o pueden ser transformados en valores cuantitativos en cualitativos, o viceversa según se considere necesario (Drew, 2008). La recolección de los datos se hace de forma automatizada con la ayuda de interfaces de programación de aplicaciones (APIs) de código abierto, que están conectadas a las fuentes de datos públicas y a través de las cuales se puede acceder a ellos. Por último, la exploración se realiza mediante

herramientas de visualización como gráficos y técnicas estadísticas con el propósito de tener un mejor entendimiento de la naturaleza de los datos

2.2.3 Preparación de los datos.

Preparar los datos significa realizar la transformación y limpieza de los mismos para conformar los conjuntos de datos que serán utilizados en la fase de modelado; algunas de las actividades en esta fase son la estandarización de formatos, eliminación de registros duplicados, la sustitución de valores no válidos, y la selección de los atributos más representativos (Ved, 2018). La importancia de preparar los datos para el análisis radica principalmente en que los datos reales suelen estar incompletos, tener errores, o ser inconsistentes.

El primer paso para la preparación de los datos, es la selección de los mismos; en este punto es necesario considerar su confiabilidad, pues de ello depende la representatividad de los resultados; enseguida se procede a limpiar los datos; usualmente este paso es el más largo, ya que se deben eliminar los registros que no son de utilidad, completar los faltantes, y estandarizar su estructura. Posteriormente se realiza la conformación de los conjuntos de datos, que se refiere a la preparación de atributos derivados o nuevos que se integran combinando diversas fuentes. Por último, se da formato a los datos según sea necesario; por ejemplo, convertir valores numéricos en valores alfanuméricos o viceversa. (Data Science Project Management). Cabe resaltar que la preparación de los datos representa alrededor del 80% del trabajo en el proyecto (Redman, 2018), y de esta depende que las siguientes etapas de la metodología arrojen resultados significativos.

2.2.3.1 Análisis de sentimientos.

En este proyecto, se realiza análisis de sentimientos sobre noticias financieras a partir del que se generan atributos calculados para conformar un nuevo conjunto de datos. El análisis de sentimientos se refiere a la aplicación de técnicas de análisis de texto, y procesamiento del lenguaje natural para identificar y estudiar las preferencias subjetivas o estado emocional expresadas en un texto (Jiawei, 2019); estos últimos describen la actitud del redactor y su sensación o valoración hacía un acontecimiento (Indurkhya & Damerau, 2010). La tarea consiste en convertir texto de lenguaje natural no estructurado en datos estructurados, y realizar una clasificación de la subjetividad, es decir catalogarlos como positivos, neutrales o negativos según sea el caso

(Prabowo & Thelwall, 2009). Para que la clasificación sea significativa, es importante considerar una colección de textos que represente la opinión de diversos individuos o entidades. El análisis parte del supuesto de que palabras como *bueno* y *excelente* indican positividad; mientras que *malo* u *horrible* indican negatividad; sin embargo, adicionalmente es necesario tomar en cuenta la semántica y el léxico, lo que hace que el análisis de sentimientos sea una tarea compleja (Indurkhya & Damerau, 2010).

Existen métodos de aprendizaje supervisado para la clasificación de sentimientos a partir de bolsas de palabras, como las máquinas de vectores de soporte (SVM) o los clasificadores de Naive Bayes; y de aprendizaje no supervisado como el *enfoque de orientación semántica* (algoritmo PMI-IR), o el *método basado en el léxico* (SentiWordNet). Estos últimos estiman el sentimiento con base en el léxico y la semántica a través de *n-gramas* y *synsets* respectivamente (Pandarachalil, Sendhilkumar, & Mahalakshmi, 2015), (Sindhu, Hussain Bukhari, & Hussain, 2017). Los *n-gramas* son secuencias simples de elementos o palabras que se agrupan para formar entidades independientes, donde *n* denota el número de elementos en la secuencia; y los *synsets* son conjuntos de palabras con el mismo significado (Princeton University, s.f.; Kumar, 2017).

El aprendizaje supervisado al utilizar bolsas de palabras tiene la desventaja de que, si existen errores de redacción o gramaticales, el clasificador puede pasar por alto palabras significativas; además, el sarcasmo y la ironía pueden ser malinterpretados; así mismo existe el riesgo de que la jerga pueda no ser reconocida; en contraste, los n-gramas son más informativos porque capturan el contexto alrededor de una palabra (University of Cincinnati, 2018). En ambos tipos de aprendizaje el propósito es identificar la polaridad del sentimiento; sin embargo, debido a la carencia de datos etiquetados, el aprendizaje no supervisado se vuelve cada vez más relevante para aplicaciones reales (Hu, Tang, Gao, & Liu, 2013). Una herramienta útil para ejecutar el análisis de sentimientos es el analizador VADER (Valence Aware Dictionary and sEntiment Reasoner) utiliza un enfoque léxico basado en reglas; es decir, toma como base las palabras y el vocabulario que previamente fueron puntuados de manera subjetiva por revisores humanos, para determinar si un texto tiene un sentimiento positivo o negativo. Una ventaja de este enfoque es la rapidez de su implementación y su eficiencia al analizar grandes cantidades de datos (DeLancey,

2020); además ofrece un rendimiento equiparable al de las máquinas de vectores de soporte (SVM) (Yalçın, 2020).

2.2.3.2 Base de datos multidimensional

El modelado multidimensional es un proceso para representar el universo de datos como hechos caracterizados por dimensiones. Los hechos son el objeto principal del análisis, y usualmente están conformados por atributos calculados; mientras que las dimensiones proporcionan el contexto para el análisis y permiten analizar los datos desde diferentes perspectivas (Laker, 2006) En este sentido, un modelo multidimensional es útil para integrar datos de diversas fuentes, razón por la que en la minería de datos se ha vuelto la base para descubrir patrones (Pedersen, 2009). El modelo multidimensional es simple y permite al usuario tener una mejor comprensión de los datos y hacer consultas de forma más eficiente. (Oracle Corporation, 2003). Una de las ventajas de utilizar una base de datos multidimensional es el rendimiento, y representación de los datos en comparación con bases de datos relacionales.

La Figura 2.2 muestra una representación del modelo multidimensional para la base de datos, cuya implementación para este proyecto se propone en MySQL Workbench 8.0.21 de Oracle; se trata del segundo sistema para la gestión de bases de datos (SGBD) más utilizado a nivel global, y tiene la ventaja de ser un software de licencia libre que proporciona herramientas para el modelado utilizando lenguaje SQL con una interfaz simple e intuitiva (Shanhong, 2020; Oracle, 2021).

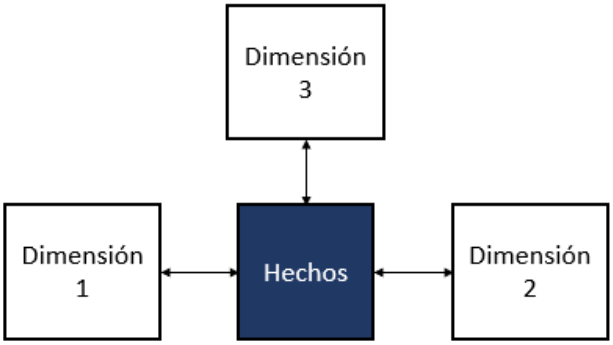


Figura 2.2. Representación de modelo multidimensional, con la tabla de hechos al centro y las tablas de dimensión alrededor. Fuente: Elaboración propia con información de (Oracle Corporation, 2003)

2.2.4 Modelado

En esta fase, se define el algoritmo de aprendizaje automático que se va a utilizar, y se genera el diseño de prueba; es decir, se define de qué forma se van a tratar los datos para entrenar el modelo; por ejemplo, separar el conjunto de datos en dos subconjuntos (entrenamiento y prueba), o en tres subconjuntos (prueba, entrenamiento, y validación). También se puede hacer una partición de los datos utilizando validación cruzada de k-iteraciones, en la que el conjunto de datos se divide en *k* subconjuntos, que de forma alternada e iterativa se utilizan como datos de entrenamiento y prueba (Amazon Web Services, Inc, 2021). Cada vez que un subconjunto *k* se utiliza como conjunto de prueba, el resto de los subconjuntos son utilizados para entrenar el modelo; de esta forma es posible probar y ajustar los parámetros a un nivel óptimo. El propósito de aplicar métodos para el modelado de los datos y técnicas de aprendizaje automático, es encontrar relaciones, identificar patrones, o hacer predicciones. Esta etapa de la metodología suele requerir de varias iteraciones entre las que se van realizando ajustes a los parámetros hasta hallar el modelo óptimo según los datos provistos (IBM Analytics, 2015).

2.2.4.1 Aprendizaje automático

El aprendizaje automático es un área de estudio que se encuentra en la intersección de las ciencias computacionales y la estadística, que se enfoca en construir sistemas computacionales capaces de hacer predicciones, tomar decisiones, y obtener conocimiento útil; cuyo rendimiento puede ser mejorado de forma automática. (Jordan & Mitchell, 2015; Mohri, Rostamizadeh, & Talwalkar, 2018). Algunas de las aplicaciones del aprendizaje automático que han sido ampliamente estudiadas son *clasificación, regresión, agrupación,* y *reducción dimensional;* y de acuerdo con el tipo, se catalogan como *aprendizaje supervisado, no-supervisado, semi-supervisado, y reforzado* (Mohri, Rostamizadeh, & Talwalkar, 2018).

Las técnicas de aprendizaje automático supervisado requieren que al modelo le sean proporcionados datos previamente etiquetados, a manera de que aprenda de conocimiento existente, y sea capaz de hacer predicciones con base en ella. Entre los algoritmos de aprendizaje automático supervisado se encuentran las redes neuronales artificiales, cuya inspiración surge de la neurociencia y son una generalización de modelos matemáticos de sistemas nerviosos

biológicos. Cabe mencionar que las redes neuronales artificiales tienen aplicaciones para la *agrupación* mediante aprendizaje no supervisado; sin embargo, para los fines del presente proyecto se considera únicamente su aplicación con técnicas de aprendizaje supervisado.

En la Figura 2.3 se representa el elemento principal de procesamiento que es la neurona o nodo que, por medio de una función de activación, y a partir de ponderar las señales de entrada, calcula la señal de salida (Abraham, 2005); esta última puede ser la señal de entrada de otra neurona y consecuentemente puede formar una red de procesadores simples con la capacidad de resolver problemas complejos.

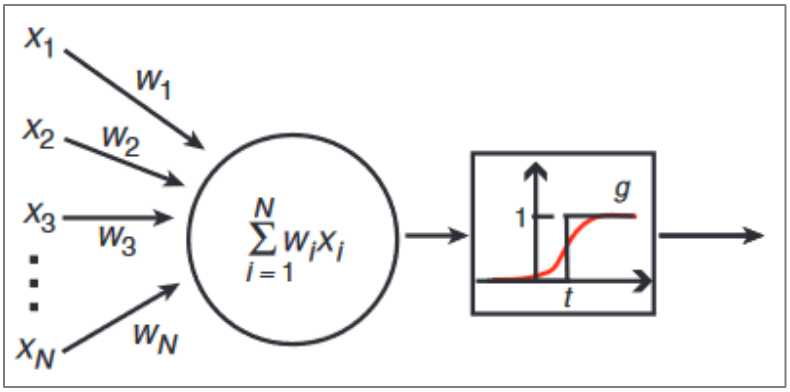


Figura 2.3. Representación de la unidad básica de procesamiento de una red neuronal artificial, por (Krogh, 2008). Cada una de las señales de entrada x1, x2, x3, xN tiene un peso definido según su relevancia; la neurona procesa las señales de entrada como una suma ponderada y utiliza el umbral de la señal de activación para determinar si emite una señal de salida.

La red neuronal más elemental, está compuesta por una sola neurona, es decir por una sola capa, y es llamada *perceptrón* (Abraham, 2005)*;* sin embargo, cuando los datos no son separables de forma lineal, es de utilidad una red neuronal con más de una capa (Krogh, 2008). La Figura 2.4 muestra una representación de una red neuronal con una sola capa oculta.

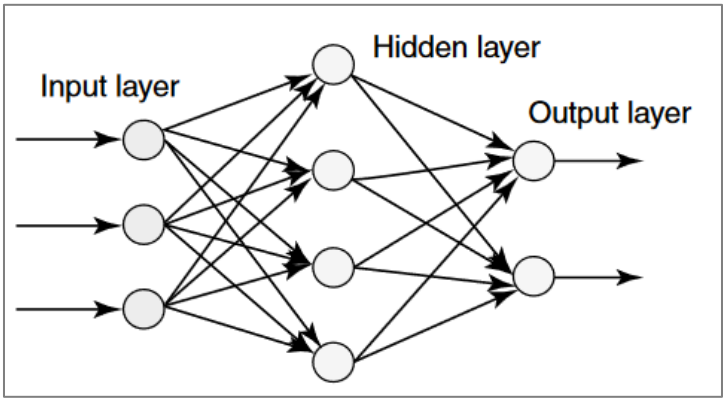


Figura 2.4 Representación de una red neuronal artificia con una capa oculta (Abraham, 2005).

La simulación del aprendizaje o *entrenamiento* en una red neuronal implica hacer pequeños cambios a los parámetros del modelo; cada vez que se presentan nuevos valores de entrada, se calcula la variación de dichos parámetros con base en la diferencia entre el valor de salida y el valor esperado (Krogh, 2008); para realizar dicho entrenamiento se utilizan métodos de optimización, entre los que se encuentran la retropropagación (*Backpropagation*) y los algoritmos genéticos (*Genetic Algorithms)*. El primero, es un algoritmo de búsqueda por descenso de gradiente que disminuye el error medio cuadrado total (Siddique & Tokhi, 2001), (Mahmood, 2019) y ayuda a que la red sea cada vez más precisa; se caracteriza por que su velocidad y robustez son sensibles a parámetros como el factor de aprendizaje, momentum, y la constante de aceleración, cuyo valor óptimo varía según el problema que se estudia (Siddique & Tokhi, 2001). El segundo, es un algoritmo de búsqueda aleatoria basado en características de la evolución biológica en el que, a partir de una población inicial, los individuos más aptos heredan sus características a la siguiente generación; y tal como sucede en la selección natural, los individuos con las mejores características tienen mayor posibilidad de sobrevivir (Mallawaarachchi, 2017). En ambos casos, cuando los cambios en el error o en características heredadas respectivamente dejan de ser significativos, termina el entrenamiento. De acuerdo con investigaciones previas (Örkcü & Bal, 2011), (Gupta & Sexton, 1999), los algoritmos genéticos ofrecen ventajas significativas en comparación con los métodos de retropropagación, una de ellas es que es menos probable que el modelo converja en mínimos locales; en contraste, tienen la desventaja de una tasa de convergencia lenta, (Zhen-Guo, Tzu-An, & Zhen-Hua, 2011), (Khan & Sahai, 2012) por lo que su selección para entrenar la red

neuronal debe considerar las características del problema que se desea resolver (Chukwuchekwa Ulumma, 2011).

Por otra parte, las técnicas de aprendizaje automático no supervisado tienen el propósito de modelar la distribución subyacente de los datos para aprender sobre ellos y descubrir patrones e información, por lo que los datos que se proporcionan al modelo no están etiquetados. Un algoritmo de aprendizaje automático no supervisado ampliamente utilizado es la *agrupación por k-medias;* cuyo objetivo es agrupar datos con características similares en *k* conjuntos. Para cada uno de los conjuntos se determina un centroide que sirve como base para determinar qué datos forman parte de él; un dato pertenece a un clúster siempre que su distancia sea la menor comparada con el resto de los conjuntos (Trevino, 2016). Como se muestra en la *Figura 2.5*, inicialmente se definen los centroides de forma aleatoria, y a partir de ellos se calcula la distancia euclidiana de cada uno de los datos hacia los centroides; una vez que se determinan los datos que pertenecen a cada centroide, se actualizan los centroides y se repite el proceso hasta que ningún dato se reasigne a otro conjunto.

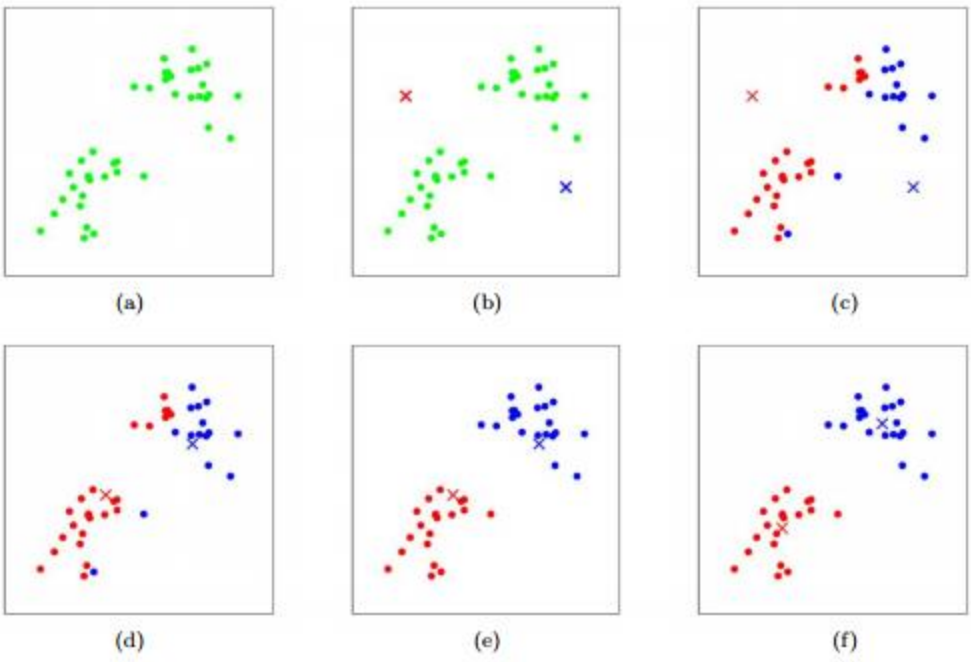


Figura 2.5. Representación del algoritmo de k-medias: (a) Conjunto de datos original. (b) Centroides aleatorios. (c-f) Ilustración del proceso para determinar las agrupaciones de datos. (Piech, 2013).

Un método para evaluar los algoritmos de aprendizaje automático es la *validación cruzada de k iteraciones.* Es un método estadístico para evaluar algoritmos de aprendizaje automático mediante la división del conjunto de datos en dos subconjuntos disjuntos por muestreo aleatorio sin remplazamiento El valor de *k* se refiere al número de subconjuntos. Para el entrenamiento del modelo se utilizan *k-1* subconjuntos, y la validación se hace con el subconjunto restante. Como se muestra en la Figura 2.6, el proceso se repite hasta que cada uno de los *k* subconjuntos ha sido utilizado para hacer la validación; de esta forma, el promedio del rendimiento en cada iteración representa el rendimiento de validación cruzada (Berrar, 2019; Refaeilzadeh, Tang, & Liu, 2016)

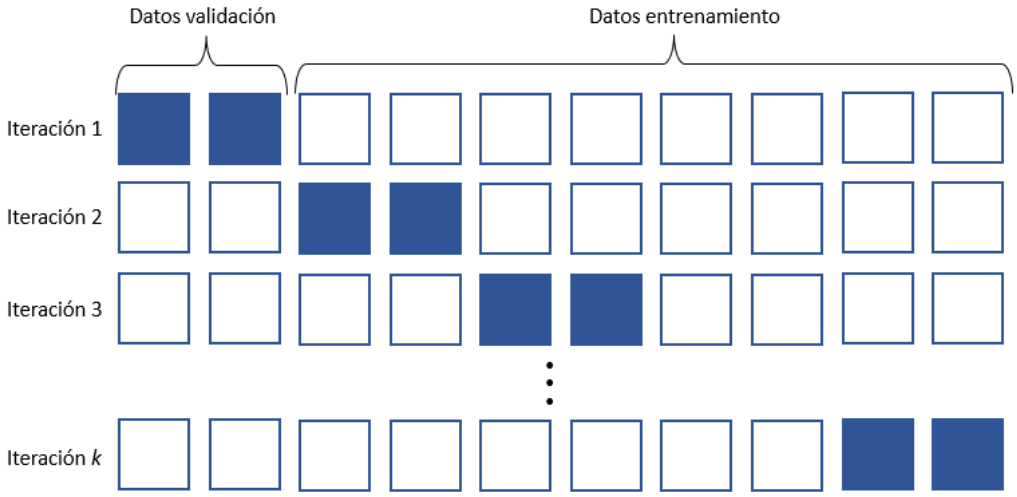


Figura 2.6. Representación gráfica del método de *validación cruzada de k iteraciones* Elaboración propia basado en (Berrar, 2019)

2.2.4.2 Herramientas para el análisis y modelado de datos.

Para realizar el análisis y modelado, los científicos de datos utilizan herramientas computacionales, entre las que se encuentran los lenguajes de programación Python y R (Pedregosa, y otros, 2011; Eastwood, 2020). Ambos de código abierto y con una amplia comunidad de usuarios que contribuye a que nuevas librerías y aplicaciones se desarrollen y se publiquen de forma recurrente. Por otra parte, existen ambientes gráficos como SAS, BigML y Amazon ML a

través de los que es posible implementar algoritmos de aprendizaje automático sin la necesidad de conocer los aspectos fundamentales de la programación. (Springboard India, 2020). Estas últimas además de que no requerir de la instalación de un software, tiene como ventaja la facilidad de uso; a pesar de ello, pueden presentar limitaciones para algunos análisis debido a que el ajuste de parámetros es limitado o complejo.

Python y R son muy similares respecto a su funcionalidad y aplicación en el área de ciencia de datos; sin embargo, Python es más popular al ser utilizado por 57% de los científicos de datos y desarrolladores, en comparación con el 31% de uso de R (Voskoglou, 2017). R está principalmente centrado en el análisis, modelado y visualización de datos; en contraste, Python es un lenguaje con aplicaciones generales que incluye herramientas para la ciencia de datos. (Eastwood, 2020). Python es un lenguaje orientado a objetos que facilita la ejecución de tareas en módulos; además posee diversas librerías para aprendizaje automático tales como: *sckit-learn, Keras* y *TensorFlow, Pandas* es la herramienta que se utiliza para el análisis y exploración de datos, y para el análisis y clasificación de sentimientos cuenta con *NLTK: Natural Language ToolKit* (Desai, 2020). NLTK es una plataforma para desarrollar proyectos en Python relacionados con el procesamiento de lenguaje natural, y que contiene paquetes de procesamiento de texto, para clasificación, análisis sintáctico, y razonamiento semántico (NLTK Project, 2020). Además, cuenta con un analizador de sentimientos (VADER) que ayuda a clasificar texto como positivo o negativo de acuerdo con su contenido. Debido a las ventajas que ofrece Python, es el sistema en el que se desarrolla el presente proyecto.

Otro software que también es utilizado para analizar datos y crear algoritmos y modelos es *Matlab*; una de sus ventajas es su lenguaje de programación, que expresa matemáticas matriciales de forma directa (The Mathworks, Inc., 2021). Ya que los modelos de redes neuronales artificiales se pueden expresar como multiplicaciones de matrices y vectores, se propone el uso de *Matlab* para realizar el modelado de redes artificiales neuronales.

2.2.5 Evaluación.

Los modelos creados en el paso anterior se evalúan para garantizar que la resolución del problema está apegada a las necesidades y objetivos planteados. La evaluación implica el cálculo

de medidas de diagnóstico que sirven como base para determinar la precisión y exactitud de los resultados (IBM Analytics, 2015). En ocasiones es útil realizar la evaluación al mismo tiempo que se construye, con el fin de encontrar los mejores parámetros para el modelo; en ese caso, es recomendable que los datos estén divididos en tres subconjuntos, datos para entrenamiento, datos para validación, y datos para prueba; o en su defecto, utilizar validación cruzada de k-iteraciones.

La evaluación del modelo depende de su naturaleza, para un modelo de clasificación se utilizan métricas como la matriz de confusión para determinar la exactitud, precisión y exhaustividad; la exactitud se refiere al número de predicciones correctas del total de predicciones, la precisión mide la probabilidad de que la predicción sea relevante, y la exhaustividad indica la probabilidad de que una predicción relevante sea clasificada como tal (Jordan J. , 2017). Por otra parte, si el modelo es de regresión, algunas de las métricas utilizadas son la varianza explicada, el error cuadrado medio y el coeficiente de determinación (R^2), que miden la varianza en las predicciones, el nivel de ajuste del modelo a los datos, y el porcentaje de la variación de la variable dependiente que es explicada por la regresión respectivamente (Songhao, 2020).

Un factor a tomar en cuenta durante la fase de evaluación, es el ajuste del modelo; un modelo presenta subajuste cuando su rendimiento es deficiente con los datos de entrenamiento, y su capacidad de aprendizaje es limitada; por el contrario, tiene sobreajuste cuando únicamente tiene un rendimiento adecuado con los datos de entrenamiento. Estas situaciones se pueden resolver cambiando la flexibilidad del modelo mediante el ajuste de sus parámetros, o utilizando conjuntos de datos de mayor tamaño (Amazon Web Services, 2021). En esta fase es importante detectar si existen incongruencias en los resultados esperados e identificar si es necesario realizar correcciones al modelo; así mismo se definen las áreas de mejora para futuros análisis.

2.2.6 Implementación.

Una vez que el modelo aprueba la evaluación, este se implementa a través de una herramienta que le permita al usuario acceder al conocimiento generado. La implementación puede ser tan simple como un informe, o tan complejo como una aplicación integrada a un proceso operativo (IBM Analytics, 2015). Cabe mencionar que la metodología es flexible e iterable, por lo que se puede ejecutar en un orden diferente, según las necesidades del investigador; sin embargo,

iniciar con la comprensión del negocio ayuda a mantener un objetivo claro durante el desarrollo y en consecuencia a asegurar que el análisis sea relevante y reproducible.

2.3 Antecedentes en la predicción del mercado de valores

El deseo por predecir el comportamiento del precio de las acciones se remonta al año 1902, cuando William Peter Hamilton quien fuera editor del *Wall Street Journal* desde 1902 hasta 1929, publicó 255 editoriales con predicciones para el mercado de valores basándose en la teoría de Dow que estableció el fundamento para el análisis técnico (Cowles, 1933). Años más tarde, en 1988 se publicó un estudio que analizó el efecto de diversos tipos de noticias sobre el comportamiento de las acciones en el mercado bursátil (Cutler, Poterba, & Summers, 1988); y durante esa misma década se realizaron las primeras propuestas sobre el uso de aprendizaje automático para hacer predicciones en el mercado de valores (Hawley, Johnson, & Raina, 1990); también se publicaron estudios (White, 1988) acerca de la aplicación de algoritmos de aprendizaje automático para identificar irregularidades en el movimiento del precio de acciones en el mercado; y se comenzaron a tomar en cuenta factores políticos e indicadores económicos (Kohara, Ishikawa, Fukuhara, & Nakamura, 1997) para mejorar la capacidad predictiva de los modelos. A partir de entonces, se han realizado diversas publicaciones al respecto.

Hoy en día la literatura sobre el aprendizaje automático para la predicción en el mercado de valores es amplia. Nti et al. (2019) analizaron una recopilación de 122 trabajos publicados entre 2007 y 2008 en el área de la predicción del mercado de valores utilizando aprendizaje automático. En su investigación categorizaron los estudios por estrategia, origen de los datos, y por el algoritmo de aprendizaje automático utilizado; y encontraron que las metodologías empleadas con mayor frecuencia fueron las máquinas de vectores de soporte (SVM) y redes neuronales artificiales (ANN); estas últimas son, utilizadas en mayor medida debido a que han demostrado ofrecer mejores resultados en comparación con métodos estadísticos tradicionales, y con otros algoritmos de aprendizaje automático (Yoo, Kim, & Jan, 2005). Por otra parte, respecto al análisis de sentimientos, existen diversos estudios que hicieron uso de noticias y de publicaciones en redes sociales (Twitter, Stocktwits) junto con herramientas como diccionarios de sentimientos o bolsas de palabras como base para determinar el sentimiento de los inversionistas ante instrumentos financieros (Schumaker & Chen, 2009; Baker & Wurgler, 2007). En la mayoría de los casos, las

investigaciones se enfocan en el análisis de un único instrumento bursátil y no revisan a detalle los datos técnicos; además de acuerdo con (Khan W. , y otros, 2020), la literatura que combina datos de redes sociales y noticias financieras es escasa.

Existe evidencia de estudios previos que estudian modelos de aprendizaje automático para predecir el comportamiento de acciones en el mercado de valores, y que utilizan datos de noticias financieras. (Zhai, Hsu, & Halgamuge, 2007) realizaron un estudio en el que integraron datos de noticias financieras e indicadores técnicos, a los que les aplicaron máquinas de vectores de soporte (SVM) para predecir si la variación diaria en el precio de la acción de una empresa australiana del sector minero (BHP.ax) es positiva o negativa; a partir del estudio concluyeron que la integración de ambos tipos de datos mejora significativamente el rendimiento predictivo del modelo. (Vargas, dos Anjos, Bichara, & Evsukoff, 2018) propusieron un modelo de red neuronal convolucional recurrente en el que integraron datos de indicadores del análisis técnico y noticias financieras para predecir la dirección en el cambio en el precio de las acciones de la empresa petrolera estadounidense Chevron. En 2019, (Agrawal, Khan, & Shukla, 2019) analizaron una red de gran memoria de corto plazo (LSTM) para predecir el precio de las acciones de tres empresas financieras utilizando datos transaccionales históricos e indicadores del análisis técnico. (Khan W. , y otros, 2020) publicaron un artículo en el que integraron datos de noticias financieras, datos de redes sociales, y datos transaccionales de 11 acciones en el mercado de valores para comparar el rendimiento de diversos algoritmos de aprendizaje automático para la identificación y clasificación de tendencias futuras en el mercado. En este contexto, es importante mencionar que, con excepción del estudio realizado (Khan W. , y otros, 2020), el objetivo de las publicaciones revisadas es clasificar el precio de las acciones en el mercado de valores a partir de datos transaccionales, indicadores derivados del análisis técnico, y/o datos de noticias financieras.

En relación con el tema cabe destacar que existe evidencia de que el uso de sistemas automáticos para realizar transacciones en el mercado de valores aumentó significativamente entre 2003 y 2012, resultando en un incremento del 70% en las transacciones algorítmicas. (Atkins, Niranjan, & Gerding, 2018). Esto deja ver que existe una tendencia por tomar decisiones basadas en datos, para reducir el impacto del factor psicológico. Recientemente se han hecho propuestas de modelos predictivos que utilizan algoritmos basado en redes neuronales artificiales como ELM (Extreme

Machine Learning [Aprendizaje Automático Extremo]) y RBFN (Radial Basis Function Network [Redes Neuronales de Base Radial]) (Li, y otros, 2013). El primero se caracteriza por definir valores aleatorios para el peso de las capas de entrada, mientras que lo de salida son calculados (Science Direct, 2021). El segundo por tener solo una capa oculta y por la función de activación de tipo radial (Chandradevan, 2017). También existen publicaciones en las que estudian un modelo predictivo desde el enfoque del *Big Data*, que concluyen que el tamaño del conjunto de datos tiene un impacto significativo sobre la precisión del modelo predictivo (Renault, 2020; Attigeri, Manohara, Pai, & Nayak, 2015).

La Tabla 2-1, presenta una compilación de 20 estudios e investigaciones consultados cuyo tema principal es el uso de aprendizaje automático para proponer modelos predictivos en el mercado de valores utilizando análisis de sentimientos; los artículos a que se hace referencia, son catalogados de acuerdo con la técnica de aprendizaje automático utilizada, según la fuente de datos (transaccionales, noticias), y el tipo análisis que toman como base (técnico, fundamental, combinado). En la mayoría de los artículos revisados, las técnicas de aprendizaje automático fueron aplicadas principalmente para clasificar los sentimientos de las noticias en torno a los instrumentos de inversión; adicionalmente en algunos casos utilizaron métodos estadísticos tradicionales (GARCH, ARIMA, Regresión Lineal) para complementar el análisis. En todos los estudios, se determinó que la integración de noticias contribuyó positivamente a la capacidad predictiva del modelo propuesto. Únicamente dos publicaciones, realizaron una revisión de la correlación entre instrumentos, específicamente entre índices bursátiles, materias primas y tipo de cambio; sin embargo, no consideraron en el análisis el efecto de las noticias (Shen, Jiang, & Zhang, 2012; Choudhry & Garg, 2008). Mientras que solo un estudio integró el sentimiento de noticias políticas y económicas junto con indicadores económicos (tasa de interés, tipo de cambio, precio de petróleo) para generar un modelo predictivo utilizando una red neuronal artificial; pero, a diferencia del presente proyecto, las noticias fueron recabadas de forma manual y el análisis de sentimientos se basó en el criterio del investigador (Kohara, Ishikawa, Fukuhara, & Nakamura, 1997).

A diferencia de otros estudios y publicaciones, este proyecto sigue una metodología CRISP-DM para generar e implementar un modelo predictivo del precio de las cincuenta acciones más representativas del índice bursátil S&P500, utilizando datos transaccionales de dichas acciones,

noticias financieras y económicas entorno a ellas, e indicadores económicos (tipo de cambio, índices económicos y materias primas), con el propósito de realizar un análisis bursátil combinado en el que consideran las circunstancias reales del entorno y la correlación entre los activos. El proyecto busca reducir el impacto del factor psicológico durante la toma de decisiones del inversionista y que las decisiones se tomen con base en datos mediante el uso de herramientas computacionales para la recolección, exploración, análisis y modelado de los datos.

Tabla 2-1

Resumen de publicaciones

Técnica	Fuente de datos	Análisis	Artículos
SVM	Twitter o similar Yahoo Finance o similar	Fundamental	7
SVM	No se especifica	Técnico	2
SVM	No se especifica	Combinado	2
SVM / Redes Neuronales Artificiales	Twitter o similar Yahoo Finance o similar	Fundamental	4
SVM / Redes Neuronales Artificiales	Twitter o similar Yahoo Finance o similar	Combinado	1
Redes Neuronales Artificiales	Twitter o similar Yahoo Finance o similar	Combinado	3
Otros	Twitter o similar Yahoo Finance o similar	Fundamental	1

Nota: Algunos estudios analizan algoritmos de aprendizaje automático adicionales, para los fines de clasificación se consideran únicamente máquinas de vectores de soporte y redes neuronales artificiales.

3 Desarrollo del proyecto

3.1 Diagnóstico

En promedio, en el mercado de valores de Estados Unidos se realizan 15 mil millones de transacciones diarias por un valor acumulado de cerca de 653 mil millones de dólares (Cboe Global Markets, 2021). Uno de los índices utilizados como indicador de las acciones estadounidenses de gran capitalización es el S&P500; este está conformado por las 500 empresas más grandes y cubre aproximadamente el 80% de la capitalización del mercado estadounidense (S&P Dow Jones Indices, 2021). En contraste con el mercado estadounidense, en México se realizan en promedio 308 mil transacciones por día en el mercado de capitales; y el índice equivalente al S&P500 en el mercado de valores mexicanos es el Índice de Precios y Cotizaciones (IPC) que enlista las 35 empresas más grandes del mercado (Banco de México, 2021; BMV, 2019). A pesar del bajo volumen de operaciones; algunas acciones del mercado estadounidense se pueden transaccionar en el mercado bursátil mexicano a través del Sistema Internacional de Cotizaciones (SIC), una plataforma que permite invertir en acciones listadas en otras partes del mundo.

Este trabajo estudia el comportamiento de las 50 acciones más grandes del S&P500, que representan cerca del 42% de la capitalización del mercado, las más representativas se muestran en la Tabla 3-1 por medio de su clasificación de acuerdo al sector en el que se encuentran

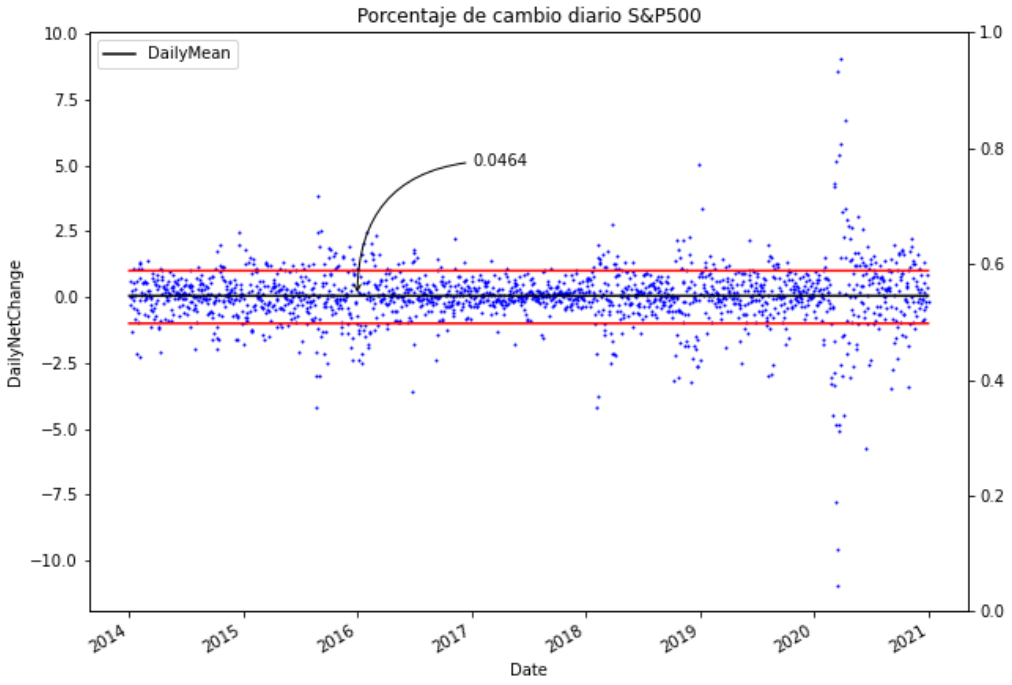
Tabla 3-1
50 acciones más representativas del S&P500 por sector.

Sector	Acciones
Tecnologías de la información	Apple Inc. [AAPL], Microsoft Corporation [MSFT], Visa Inc. Class A [V], Mastercard Incorporated Class A [MA], NVIDIA Corporation [NVDA], PayPal Holdings Inc. [PYPL], Intel Corporation [INTC], Adobe Inc. [ADBE], , Cisco Systems Inc, [CSCO], Salesforce.com Inc. [CRM], Broadcom Inc. [AVGO], Accenture Plc Class A [ACN], Texas Instruments Incorporated [TXN], Qualcomm Inc. [QCOM]
Consumo discrecional	Amazon.com Inc. [AMZN], Tesla [TSLA], Home Depot Inc. [HD], Nike Inc. Class B [NKE], McDonald’s Corporation [MCD], Costco Wholesale Corporation [COST]

Servicios de comunicación	Facebook Inc. [FB], Alphabet Inc. Class A [GOOGL], Alphabet Inc. Class A [GOOG], Walt Disney Company [DIS], Comcast Corporation Class A [CMCSA], Netflix Inc. [NFLX], Verizon Communications Inc. [VZ], AT&T Inc. [T],
Financieras	Berkshire Hathaway Inc. Class B [BRK.B], JPMorgan Chase & Co [JPM], Bank of America Corp [BAC], Citigroup Inc. [C]
Salud	Johnson & Johnson [JNJ], UnitedHealth Group Incorporated [UNH], Pfizer Inc. [PFE], Abbott Laboratories [ABT], AbbVie Inc. [ABBV], Merck & Co. Inc. [MRK], Thermo Fisher Scientific Inc. [TMO], Eli Lilly and Company [LLY], Well Fargo & Company [WFC], Medtronic Plc [MDT],
Productos básicos	Procter & Gamble Company [PG], Coca-Cola Company [KO], PepsiCo Inc. [PEP], Walmart Inc. [WMT]
Energía	Exxon Mobil Corporation [XOM]; Chevron Corporation [CVX], NextEra Energy Inc. [NEE],
Industriales	Honeywell International Inc. [HON],

Nota: Elaboración propia con datos de (State Street Global Advisors, 2021).

En este proyecto se analizan los datos de las acciones de las 50 empresas principales durante el periodo comprendido entre 2014 y 2020. En ese periodo, el porcentaje de cambio diario promedio en el índice S&P500 fue 0.0464%; a partir de lo que se puede interpretar que el índice tuvo un rendimiento promedio anual del 12.06% (Yahoo Finance, 2021). Como se puede apreciar en la Gráfica 3.1, la mayoría de las variaciones diarias del índice se encuentran entre -1% y 1%, aunque cabe destacar que existen valores atípicos, que se encuentran por fuera de ese rango; estos se pueden explicar por acontecimientos económicos o financieros destacados. Con base en la afirmación previa, los periodos asociados a los valores atípicos se encuentran principalmente en los años 2015, 2016, 2018, 2019 y 2020. Se observa que, en 2017 el porcentaje de variación diaria se mantiene entre -1% y 1%. La Tabla 3-2 resume los eventos económicos más importantes ocurridos durante el periodo estudiado; entre ellos destaca la volatilidad en el precio del petróleo, decisiones políticas y fiscales en EE. UU, relaciones internacionales entre las mayores economías mundiales.



Gráfica 3.1. Gráfico del porcentaje de cambio diario en el precio del índice S&P500 de 2014 a 2020. Elaboración propia con información de Yahoo Finance.

Tabla 3-2

Eventos económicos entre 2014 y 2020

Año	Eventos
2014	Crisis en Ucrania y Rusia; volatilidad en el precio del petróleo. (Walker, 2014)
2015	China emerge como la mayor economía mundial; Suiza elimina tipo de cambio mínimo frente al euro; cambios en tasa de interés en Australia, China y EE.UU. (Vessiari, 2016)
2016	Brexit; elección del presidente Donald Trump; volatilidad en el precio del petróleo; preocupación por desaceleración económica de China; incremento en tasa de interés de EE.UU. (Sullivan, 2016)
2017	Recorte de impuestos en EE.UU.; popularidad de las criptomonedas (Dorman, 2017)
2018	Guerra comercial entre China y EE.UU.; nivel de desempleo más bajo en EE.UU. desde 1969; tratado entre México, EE.UU. y Canadá (T-MEC) (Carson, 2018)
2019	Desaceleración económica global; tensiones entre EE.UU. y China (Gopinath, Milesi-Ferretti, & Nabar, 2019)

Año	Eventos
2020	COVID-19; baja demanda de petróleo a nivel global; estímulos fiscales y elecciones presidenciales en EE.UU. (Ausenbaugh, Faller, & Cohen, 2020)

Nota: Los eventos están centrados en EE.UU. en relación con el índice S&P500

Los eventos económicos suelen ser detonadores para la toma de decisiones por parte de los inversionistas; dependiendo del contexto, los acontecimientos pueden ser interpretados como positivos o negativos; y, con relación a la aseveración de las finanzas conductuales, los inversionistas pueden reaccionar de forma irracional. El exceso de confianza, y el comportamiento gregario son factores que influencian las decisiones del inversionista; estudios empíricos proporcionan evidencia de que los inversionistas generan un exceso de confianza cuando han tenido rendimientos positivos en inversiones previas; y que ante eventos financieros negativos tienden a tener un comportamiento gregario independientemente de la rentabilidad del mercado (Alsabban & Alarfaj, 2019; Economoua, Hassapiscand, & Philippa, 2018). La reacción del inversionista ante los eventos económicos es más pronunciada cuando el sentimiento es negativo; sin embrago, también ha sido demostrado que el profesionalismo y la experiencia reducen los sesgos cognitivos que llevan a tomar decisiones irracionales (Kudryavtsev, Cohen, & Hon-Snir, 2013; Schmeling, 2007)

Entre los eventos económicos listados en la Tabla 3-2 destaca la volatilidad en el precio del petróleo, y la relación económica global con China; en ese sentido vale la pena mencionar que existe una correlación negativa entre el precio del petróleo y el sentimiento de inversión; es decir, cuando el precio del petróleo cambia de forma abrupta, la percepción positiva del inversionista reduce (Apergis, Cooray, & Rehman, 2017). Además, durante periodos de crisis financiera existe una dependencia positiva entre el mercado accionario y el mercado de petróleo, que conduce a que su comportamiento sea homologo. (Bampinas & Panagiotidis, 2017). En relación a los eventos acontecidos en el periodo de estudio y a la correlación que existe entre indicadores económicos y el comportamiento del mercado bursátil; la Tabla 3-3 contiene los índices bursátiles considerados en este proyecto, y su representación como indicadores económicos. Es importante mencionar que activos como el petróleo o divisas son consideradas en este estudio debido a que en estudios previos

se ha planteado que existe una dependencia entre ellos y el mercado de valores (Ergeshidze, 2017; Miller & Ratti, 2009).

Tabla 3-3

Índices bursátiles según el indicador económico que representan

Índice [*código bursátil*]	Indicador económico
Volatilidad [*VIX*]	Mide el nivel de incertidumbre en el mercado basado en expectativas del comportamiento del S&P500. Valores elevados indican mayor grado de incertidumbre (desconfianza) (Cboe Exchange, Inc., 2021)
MSCI Mercados Emergentes [*EEM*]	Desarrollo económico de mercados emergentes a nivel global (MSCI Inc., 2021)
Interés de los bonos de la tesorería a 10 años [*TNX*]	Valora la confianza de los inversionistas hacia el crecimiento económico. (Amadeo, 2021)
Petróleo crudo [*CL=F*]	Precio del petróleo a futuro (Miller & Ratti, 2009) (Kayalar, Küçüközmen, & Selcuk-Kestel, 2017)
Futuros del S&P500 [ES=F]	Mide las expectativas sobre el valor del índice S&P500 a futuro (Ross, 2020).
USD/MXN [*MXN=X*]	Tipo de cambio (Ergeshidze, 2017)
USD/EUR [*EUR=X*]	Tipo de cambio (Ergeshidze, 2017)

3.2 Metodología

Tomando como base la metodología de minería de datos CRISP-DM, este proyecto sigue el proceso esquematizado en la Figura 3.1. El primer paso es comprender el problema que se desea resolver. Enseguida se procede a explorar y preparar los datos; como se mencionó en secciones anteriores, este paso es el más largo y puede representar alrededor del 80% del trabajo. En el tercer paso se genera y evalúa un modelo predictivo, el cual, en el último paso, se implementa a través de un reporte de la predicción en el cambio del precio de las acciones estudiadas. La metodología

CRISP-DM no termina en el último paso, el ciclo puede ejecutarse tantas veces como se considere necesario.

El proyecto se desarrolla utilizando el lenguaje de programación *Python* a través de la aplicación web de código abierto *Jupyter Notebooks* que proporciona un ambiente interactivo para la visualización y análisis de datos (Project Jupyter, 2021). La razón de utilizar *Python* es que es un lenguaje de programación con una amplia comunidad de usuarios, por lo que la disponibilidad de herramientas para la recolección, visualización, preparación y análisis de los datos es extensa. Este proyecto se lleva a cabo utilizando librerías y paquetes de uso libre que proporcionan herramientas para la recolección de datos, limpieza, transformación y modelado de acuerdo con las fases planteadas en la metodología CRISP-DM.

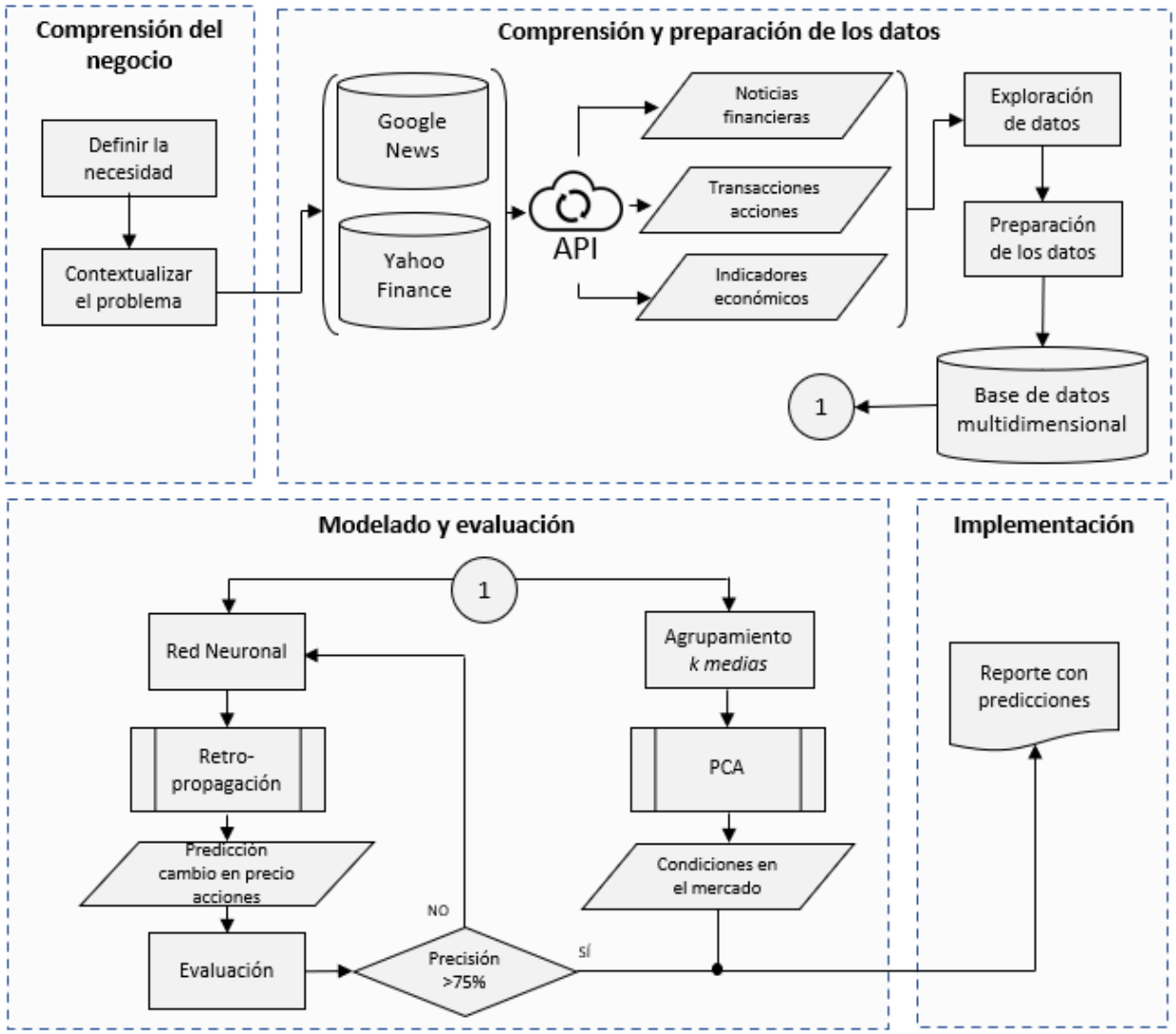


Figura 3.1 Proceso para el desarrollo del proyecto, basado en la metodología CRISP-DM. Elaboración propia con información de (IBM Analytics, 2015)

3.2.1 Comprensión del negocio

Esta fase tiene como propósito alinear los objetivos del proyecto con las necesidades que busca resolver; en otras palabras, se trata de contextualizar el objetivo general para que el desarrollo e implementación del proyecto se ejecuten en congruencia. Para ello, esta fase se lleva a cabo tomando como base la metodología *challenge-driven innovation* planteada por la empresa *Innocentive*, que consiste en cuatro pasos:

1. Establecer la necesidad de la solución: El inversionista está expuesto a riesgos inherentes al mercado de valores, que pueden resultar en pérdidas económicas; cuando una inversión representa un mayor riesgo, existe la posibilidad de generar mayores ganancias; sin embargo, este hecho favorece la toma de decisiones irracionales basadas en emociones (López-Cabarcos, Pérez-Pico, Vázquez-Rodríguez, & López-Pérez, 2019)

2. Justificar la necesidad: Como se ha mencionado, los inversionistas no siempre se comportan de forma racional y, en su lugar, actúan basados en sus emociones; esto propicia que se puedan ver influenciados por factores conductuales y sociales que conducen a la toma de decisiones de forma precipitada, y que tienen como posible consecuencia una reducción en el rendimiento de una inversión. En este sentido, este proyecto aporta a mejorar la gestión de los activos financieros de forma indirecta al disminuir la subjetividad en la toma de decisiones. Por otra parte, es un antecedente para la aplicación de metodologías de aprendizaje automático, minería de textos y análisis de sentimientos, en la toma de decisiones respecto a un grupo de acciones.

3. Contextualizar el problema: De acuerdo con la revisión de la literatura, existen diversos estudios y propuestas en relación con la generación de modelos predictivos para acciones en el mercado bursátil. De forma general, los estudios que integraron datos de noticias financieras o indicadores bursátiles y económicos llegaron a la conclusión de que la integración de dichos datos contribuye a mejorar la precisión de los modelos predictivos. También es notorio que la gran mayoría de los estudios revisados se enfocan principalmente en la aplicación de técnicas de aprendizaje automático para clasificar los sentimientos de las noticias en torno a los instrumentos de inversión. Este proyecto se diferencia del resto en tres aspectos: metodología de implementación CRISP-DM, integración de tres tipos de datos (noticias financieras, datos transaccionales de las acciones, e indicadores económicos), y el tipo de algoritmos de aprendizaje aplicados.

4. Redactar el planteamiento del problema: Dado que en el mercado bursátil existen factores que favorecen la toma de decisiones de manera irracional, el inversionista se enfrenta a la incertidumbre de la correcta gestión de los instrumentos de inversión y por lo tanto su

rendimiento se puede ver afectado de forma negativa. Para reducir la incertidumbre se propone un modelo predictivo que integra datos de noticias financiera, índices económicos, y datos transaccionales de las acciones, y sirve al inversionista como herramienta de soporte al inversionista al momento de tomar decisiones de inversión.

3.2.2 Comprensión y preparación de los datos

Previo a comprender y preparar los datos, se realiza la recolección de los datos; esta se hace utilizando interfaces de programación de aplicaciones (API), a través de las que se establece comunicación con las fuentes de datos. Como se muestra en la Figura 3.1, los datos son recolectados de dos fuentes: *Google News* para las noticias financieras, y *Yahoo Finance* para los datos transaccionales de las acciones, e indicadores económicos. Las APIs están disponibles en el repositorio de software para Python *Python Package Index* (PyPI)[1] ; en este caso se accede a ellas a través de Jupyter Notebooks

3.2.2.1 Recopilación de los datos

La recopilación de las noticias financieras, y datos transaccionales en torno a las acciones que se estudian en este proyecto se realiza utilizando una API para acceder a las bases de datos de *Google News y Yahoo Finance*. Se define un rango de fechas entre el 01 de enero de 2014 y el 31 de diciembre de 2020 que corresponde a 2,556 días; y una periodicidad diaria. El método utilizado para recolectar los datos es *Web Scraping,* la cual es una técnica para extraer el contenido de un sitio web, y almacenarlos en un archivo o una base de datos (Zhao, 2017); el proceso se puede ejecutar de forma manual o automática. En el caso de este proyecto se realiza de forma automática a través de APIs con la ventaja de que las consultas se realizan con mayor velocidad y por tanto los datos se obtienen en menor tiempo. Sin embargo, esto puede causar que la dirección IP (Internet Protocol) sea bloqueada por el sitio web. Particularmente, el sitio *GoogleNews* previene que se hagan consultas cuando detecta una actividad anormal, y bloquea la dirección IP del cliente. Actualmente no existen normativas claras referentes al *Web Scraping,* y tomando como marco de referencia el trabajo de Krotov & Silva (2018) en el que hacen una revisión de la legalidad y ética

[1] https://pypi.org/project/yfinance/; https://pypi.org/project/GoogleNews/

del *Web Scraping*, se determina que la aplicación de la técnica en este proyecto no infringe los términos de uso, ni derechos intelectuales; además, los datos recolectados no son privados, y no se causa daño a los sitios durante el proceso.

Con el propósito de evitar el bloqueo de la dirección IP, se utiliza un servidor proxy con direcciones rotativas. El servidor asigna una dirección diferente cada vez que se establece una conexión al sitio web del que se recopilan los datos; de ese modo, las consultas se realizan de forma indirecta, y el sitio web no puede identificar que se trata de un proceso automático de recopilación de datos. La Figura 3.2 muestra el proceso de forma esquemática y simplificada; el servidor proxy es un elemento intermediario entre el cliente y el sitio web, cuya función es identificar al cliente que realiza consultas de manera inusual; desde la perspectiva del sitio web, cada consulta es hecha por un usuario (dirección IP) diferente. Como he mencionado el proyecto se desarrolla utilizando Python; y la recopilación de los datos se lleva a cabo con ayuda de librerías de código abierto: *sys*[2], *ssl*[3], *urllib.request*[4], para acceder a variables y funciones del compilador, proveer accesos al protocolo de red SSL (Secure Sockets Layer)[5], y acceder a sitios web respectivamente; también se utilizan librerías para definir formatos de datos (date, datetime). Adicionalmente, se usan APIs y un servidor proxy; este último es un servicio de suscripción mensual proporcionado por la empresa NordVPN cuya implementación es posible en Python mediante la liberaría *nordvpn-switcher*[6].

[2] Permite el acceso a parámetros y funciones específicas del sistema Python (Python, 2021)
[3] Permite el acceso a sitios encriptados por SSL (Secure Sockets Layer) (Python, 2021)
[4] Librería que facilita el acceso a sitios web a través de la URL (Uniform Resource Locator) (Python, 2021)
[5] Protocolo para establecer una conexión autentificada y encriptada entre computadoras conectadas a una red (SSL Support Team, 2019)
[6] Rotación de servidores proxy de VPN (Boghe, 2021)

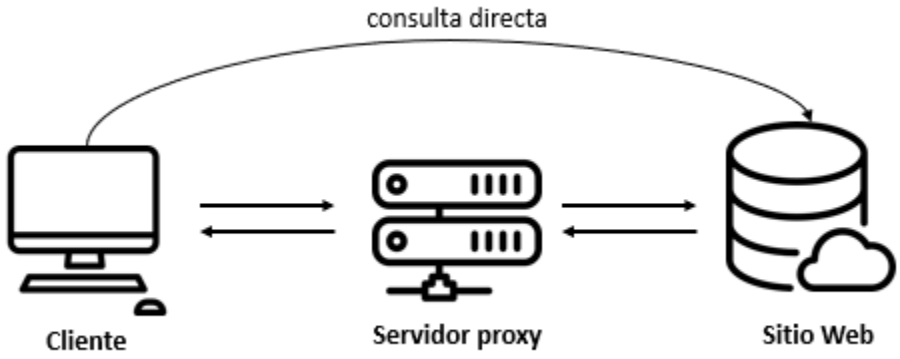


Figura 3.2 Proceso de recolección de datos utilizando un servidor proxy con direcciones rotativas. Elaboración propia

Recopilación de datos de noticias financieras de Google News. Para la consulta de los datos se proporciona el concepto de búsqueda, en este caso el nombre de las acciones junto con su código bursátil[7]; y el rango de fechas. A partir de esto, se obtiene un conjunto de datos específicos de cada acción por día. El conjunto de datos de *GoogleNews* está conformado por las variables: *title, media, date, datetime, desc, link, img*. Se asigna el código bursátil (*ticker)* correspondiente a cada registro, con el fin de poder identificarlos; así, los atributos que sirven como llave para relacionar los conjuntos de datos son *date* y *ticker*. La Tabla 3-4 muestra un ejemplo de los datos recolectados; y el código correspondiente implementado en *Python* se encuentra en la sección 6.1.

Tabla 3-4

Datos de noticias financieras recopilados de Google News

Atributo	Valor	Fuente
title	Mac Pro, iMac & Qualcomm: What to expect from Apple in the start of 2019	Google News
media	AppleInsider	Google News
date	Jan 1, 2019	Google News
datetime	01/01/2019 12:00:00 a. m.	Google News

[7] El código bursátil de las acciones se puede consultar de forma pública en: https://www.nasdaq.com/market-activity/stocks/screener

desc	A new "pro display" was also promised, to accompany what Apple marketing chief Phil Schiller advised would be Apple's "highest-end, high-throughput	Google News
link	https://appleinsider.com/articles/19/01/01/mac-pro-imac-qualcomm-what-to-expect-from-apple-in-the-	Google News
img	yH5BAEKAAEALAAAAAABAAEAAAICTAEAO	Google News
ticker	AAPL	-

Nota: Ejemplo de los datos de las noticias financieras recopiladas de Google News.

Recopilación de datos transaccionales de *Yahoo Finance*. Para recolectar los datos transaccionales de *Yahoo Finance*, se utiliza la herramienta *yfinance*[8], de uso libre disponible en el repositorio de librerías para Python (PyPI). Como criterios de entrada para la recolección, se proporciona el código bursátil de las acciones, así como el periodo; con esto se obtiene un conjunto de datos conformado por los atributos: *date, open, high, low, close, adj. close, volumen.* Con el fin de identificar a qué acción corresponde cada registro, se asigna el atributo *ticker* que corresponde al código bursátil de la acción. En la
Tabla 3-5 se muestra un ejemplo de los datos recolectados; el código correspondiente implementado en *Python* se presenta en la sección 6.2.

Tabla 3-5

Datos transaccionales de las acciones, recopilados de Yahoo Finance

Atributo	Valor	Fuente
date	02/01/2019	Yahoo Finance
open	38.7224998	Yahoo Finance
high	39.7125015	Yahoo Finance
low	38.5574989	Yahoo Finance
close	39.4799995	Yahoo Finance

[8] Librería para la recopilación de datos históricos de *Yahoo Finance.*

adj. close	38.505024	Yahoo Finance
volume	148158800	Yahoo Finance
ticker	AAPL	-

Nota: Ejemplo de los datos transaccionales de las acciones recopiladas de Yahoo Finance.

Recopilación de índices bursátiles de *Yahoo Finance*. Por último, haciendo uso de la librería *yfinance* se hace la recolección de los datos de los índices bursátiles indicados en la Tabla 3-3. Los criterios de entrada en este caso son los códigos bursátiles de los índices, y el rango de fechas del que se desea obtener los datos. El conjunto de datos obtenido se muestra en la Tabla 3-6, y está conformado por los atributos: *date, high, low, close;* se agrega el atributo *ticker* a cada registro para identificar el índice bursátil correspondiente. El código implementado en *Python* para la recolección de los datos se presenta en la sección 6.2.

Tabla 3-6

Datos transaccionales de las acciones, recopilados de Yahoo Finance

Atributo	Valor	Fuente
date	02/01/2019	Yahoo Finance
high	28.5300007	Yahoo Finance
low	23.0499999	Yahoo Finance
close	23.2199993	Yahoo Finance
ticker	^VIX	-

Nota: Ejemplo de los datos de los índices bursátiles obtenidos de *Yahoo Finance*.

Tras la recopilación de los datos en el periodo de estudio se obtienen tres conjuntos: 1) noticias financieras, 2) datos transaccionales, e 3) indicadores económicos. El total de noticias para las 50 acciones resulta en 397,412 registros. Cabe mencionar que el número de noticias diarias para cada acción es variable, pues depende de la cobertura que hayan realizado los medios de comunicación; en este sentido, pueden existir días en que no se hayan publicado noticias o, por el contrario, que exista más de una noticia para cada acción. El número de registros de los datos transaccionales de las acciones e indicadores económicos es de 87,625 y 14,117 respectivamente, como se muestra en la Tabla 3-7.

Tabla 3-7

Características de las fuentes de datos

Conjunto de datos	Fuente de datos	Número de atributos/variables	Número de registros
1	Google News	8	397,412
2	Yahoo Finance [transacciones]	8	87,625
3	Yahoo Finance [indicadores]	5	14,117

3.3 Exploración y preparación de los datos

3.3.1 Exploración de datos de *Google News*

Una vez que los datos fueron recolectados, se procede a explorarlos, con el fin de identificar anomalías como datos faltantes o duplicados; o tipos de datos incongruentes, etcétera. En el caso de los datos de las noticias financieras obtenidas de *Google News* se detectó que algunos registros se encuentran en idiomas diferentes al inglés (idioma por defecto); por lo que se procede a identificarlos y eliminarlas del conjunto. La identificación del idioma de cada noticia se hace de forma automática utilizando una librería de uso libre *langdetect* y asignando un nuevo atributo *lang* a cada registro, como se muestra a continuación:

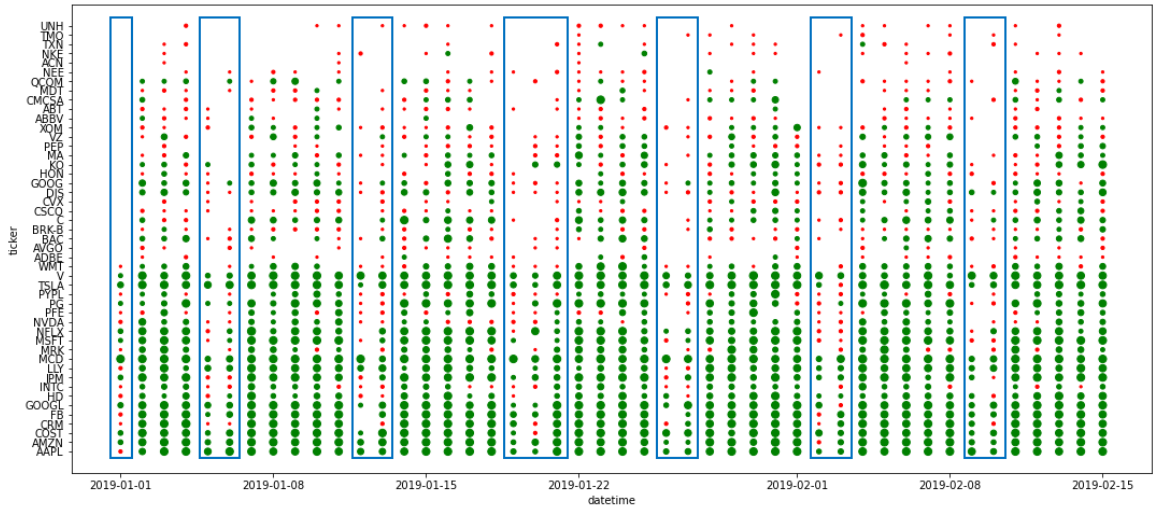
```
#Identificar el idioma de cada noticia
# Adaptación realizada a partir de una publicación realizada por Danilak, Michal el 05/03/2020
# en el sitio: https://pypi.org/project/langdetect/
##
data = pd.read_csv("news2.csv",engine='python')
data['lang']= 'null'
for i in range(len(data['title'])):
    try:
        data['lang'][i] = detect(data['title'][i])
    except:
        data['lang'][i] = 'undefined'

data_en = data[data['lang'] == 'en'] #Filtra noticias en inglés
data_en = data_en.reset_index(drop=True)
data_en.iloc[18]
```

El atributo sirve como indicador para descartar los registros que no aportan información para el análisis; de esta forma, el nuevo conjunto de datos se reduce a 381,946. Adicional a las diferencias en el lenguaje, se identificó que del conjunto de datos que contiene las noticias financieras, solo son de interés los atributos *title*, *desc*, *datatime* y *ticker*, el resto de los atributos (*media, link, img*) no aportan valor para el análisis, pues se trata del medio que publicó la noticia, el hipervínculo a la publicación e imágenes respectivamente, por lo que se toma la decisión de eliminarlos. Por último, utilizando la función *info()* de la librería *Pandas*, se obtiene un resumen del conjunto de datos, a partir del que es posible hallar errores o incongruencias que deban ser corregidos. En la Figura 3.3 se muestra que no hay registros vacíos ni datos faltantes.

```
RangeIndex: 381946 entries, 0 to 381945
Data columns (total 4 columns):
 #   Column    Non-Null Count   Dtype
---  ------    --------------   -----
 0   title     381946 non-null  object
 1   datetime  381946 non-null  datetime64[ns]
 2   desc      381946 non-null  object
 3   ticker    381946 non-null  object
```

Figura 3.3 Información del conjunto de datos de noticias financieras obtenido de Google News. Elaboración propia.

La Gráfica 3.2 muestra la distribución de las noticias de cada acción por día. En rojo están marcadas las acciones cuyo número de noticias por día es menor que tres; mientras que en verde se identifican aquellas con al menos tres noticias por día. En el gráfico se aprecia un patrón de cinco días en verde, seguido de dos días en rojo; es decir, una semana de lunes a viernes, y un fin de semana respectivamente; a partir de lo que es posible asumir que en los días sábado y domingo hay un menor flujo de noticias en torno a las acciones.



Gráfica 3.2 Diagrama de dispersión de la cantidad de noticias de cada acción por día; con fines ejemplificativos, solo se grafican los datos de las 25 acciones más representativas en un periodo de 45 días. En rojo se señalan las acciones con menos de 3 noticias por día. Elaboración propia.

Con el objetivo de determinar si las noticias tienen un impacto en el comportamiento del precio de las acciones, procedemos a transformar los datos de las noticias en valores numéricos mediante análisis de sentimientos, con lo que se asignan atributos que indican si la noticia es percibida como negativa (*neg*), positiva (*pos*) o neutral (*neu*), según el contenido textual. La herramienta para el análisis de sentimientos aplicada en este proyecto es VADER,

accesible a través de la librería NLTK (Natural Language ToolKit) en *Pyhton*. El analizador VADER clasifica una cadena de texto como positiva o negativa utilizando como base la semántica y el léxico del idioma inglés. La Tabla 3-8 muestra el resultado del análisis de sentimientos de la concatenación de las variables *title* y *desc;* con el que se genera un nuevo conjunto de datos, que en la siguiente fase del proyecto será utilizado para conformar una de las dimensiones del modelo multidimensional. El nuevo conjunto de datos está conformado por las variables *ticker, datetime, headline+desc, neg, neu, pos,* y *compound,* que corresponden respectivamente al código bursátil de la acción, fecha, título y descripción, calificación negativa, neutral, positiva, y a la suma ponderada y normalizada de las calificaciones.

En la Tabla 3-8 se aprecia que, para la muestra seleccionada, la mayoría de las noticias son catalogadas como positivas. Es posible identificar que durante el periodo graficado hay acciones (FB) con al menos una noticia negativa por semana. En relación con los resultados de la Gráfica 3.3, se aprecia que a pesar de que durante los fines de semana se publican menos noticias, en ocasiones, su ponderación (negativa o positiva) resulta ser más alta que durante la semana laboral; en futuros trabajos valdría la pena evaluar que tan representativas son las noticias publicadas en fines de semana para el precio de las acciones en el día lunes. Cabe mencionar que hay días en los que no hay publicaciones de noticas referentes a las acciones; en estos casos la herramienta asigna por defecto una ponderación neutral.

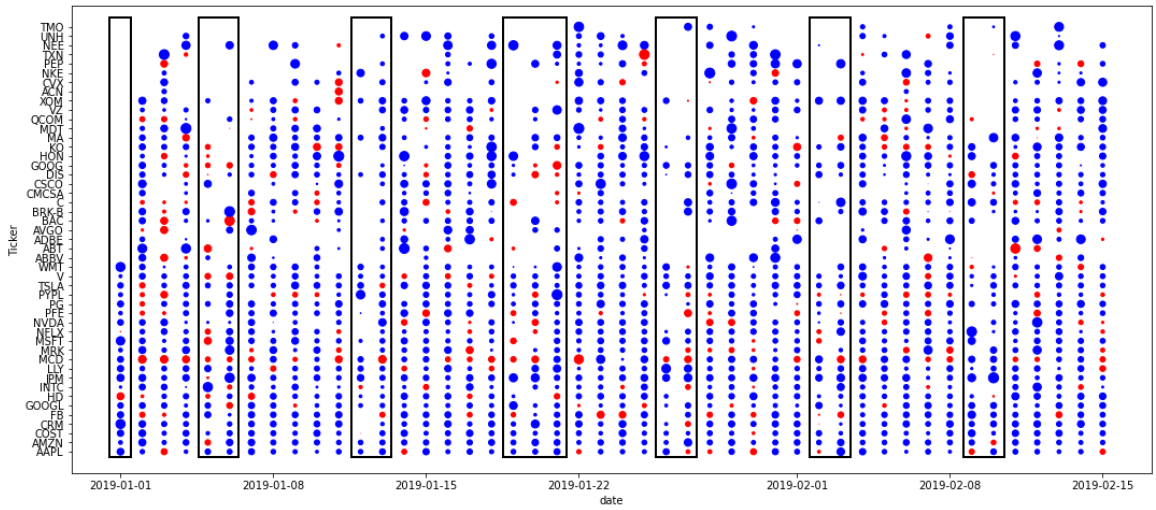
Tabla 3-8

Conjunto de datos de noticias financieras

ticker	datetime	headline+desc	neg	neu	pos	compound
AAPL	2019-01-01	Mac Pro, iMac & Qualcomm: What to expect from ...	0.000	0.933	0.067	0.3612
AAPL	2019-01-01	Bent iPad Pros, an exploded iPhone XS Max, and...	0.000	0.942	0.058	0.4019
AAPL	2019-01-01	PlayStation 4 Surpassed 90 Million Units Sold ...	0.274	0.726	0.000	-0.9274
MSFT	2019-01-01	Life Is Strange: BTS And ARK Are Now Available...	0.126	0.818	0.056	-0.2500

ticker	datetime	headline+desc	neg	neu	pos	compound
MSFT	2019-01-01	How-To Holiday Guide: How To Set Up Your New S...	0.000	0.814	0.186	0.7906

Nota: Se muestran algunos registros de forma ejemplificativa. La variable n*eg* representa el valor de negatividad que la herramienta asigna a la noticia; *neu* representa el valor de neutralidad que se asigna a la noticia; *pos* representa el valor de positividad asignado a la noticia; y *compound* representa el valor ponderado asignado a la noticia.



Gráfica 3.3 Diagrama de dispersión del sentimiento de cada noticia por día; en rojo se señalan las noticias negativas, y en azul las positivas; el tamaño de cada punto representa el grado de positividad o negatividad. Con fines ejemplificativos, solo se grafican los datos de las 25 acciones más representativas en un periodo de 45 días. Elaboración propia.

3.3.2 Exploración de datos de *Yahoo Finance*

De igual forma, los datos transaccionales de las acciones obtenidos de *Yahoo Finance* se exploran para identificar irregularidades. Lo primero es revisar que los datos estén completos y sean congruentes; mediante la función *info( )* y *describe( )* de la librería *Pandas*, como se muestra en la Figura 3.4 y Figura 3.5 no hay datos faltantes, y los valores mínimos y máximos de los precios y volúmenes de transacción se encuentran en valores positivos y conforme al comportamiento de las acciones *BAC, AMZN*, y *GOOG, AAPL* respectivamente. El origen de los datos obtenidos de *Yahoo Finance* es la empresa *Intercontinental Exchange* (ICE) que gestiona datos del mercado bursátil en tiempo real (Yahoo, s.f.); por esta razón, el conjunto de datos no presenta datos faltantes o erróneos.

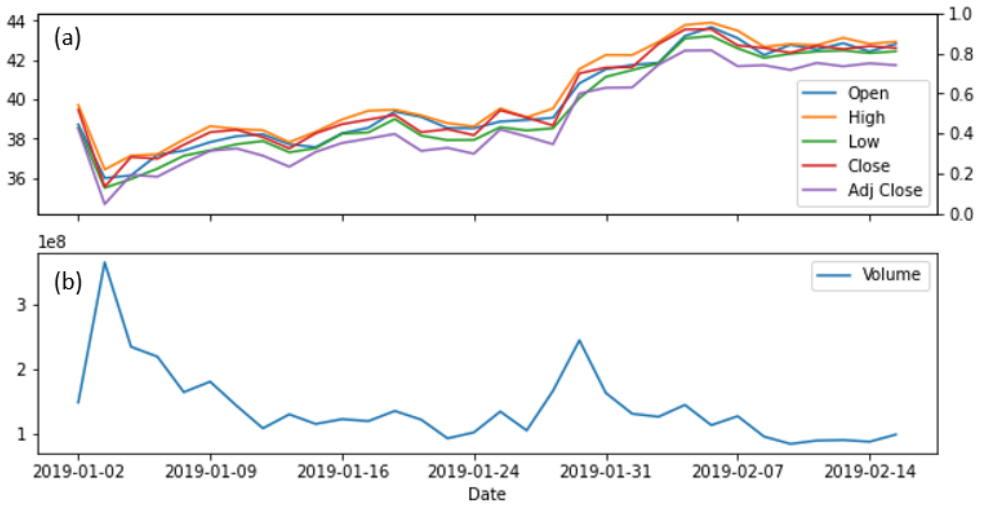
```
Int64Index: 87624 entries, 0 to 87624
Data columns (total 8 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Date       87624 non-null  object
 1   Open       87624 non-null  float64
 2   High       87624 non-null  float64
 3   Low        87624 non-null  float64
 4   Close      87624 non-null  float64
 5   Adj Close  87624 non-null  float64
 6   Volume     87624 non-null  float64
 7   Ticker     87624 non-null  object
```

Figura 3.4. Información del conjunto de datos obtenido de *Yahoo Finance*. Elaboración propia.

	Open	High	Low	Close	Adj Close	Volume
count	87624.000000	87624.000000	87624.000000	87624.000000	87624.000000	8.762400e+04
mean	158.667901	160.225622	157.027479	158.687469	153.042814	1.558771e+07
std	272.068110	274.974822	268.905766	272.049175	273.234873	2.932362e+07
min	11.460000	11.550000	10.990000	11.160000	10.022112	7.922000e+03
25%	51.900002	52.375000	51.368750	51.889999	46.235556	3.847400e+06
50%	86.362144	87.080002	85.599998	86.349998	78.291988	7.382100e+06
75%	141.899994	143.107048	140.624268	141.902496	136.157658	1.602945e+07
max	3547.000000	3552.250000	3486.689941	3531.449951	3531.449951	1.065523e+09

Figura 3.5. Descripción del conjunto de datos obtenido de *Yahoo Finance*. Elaboración propia

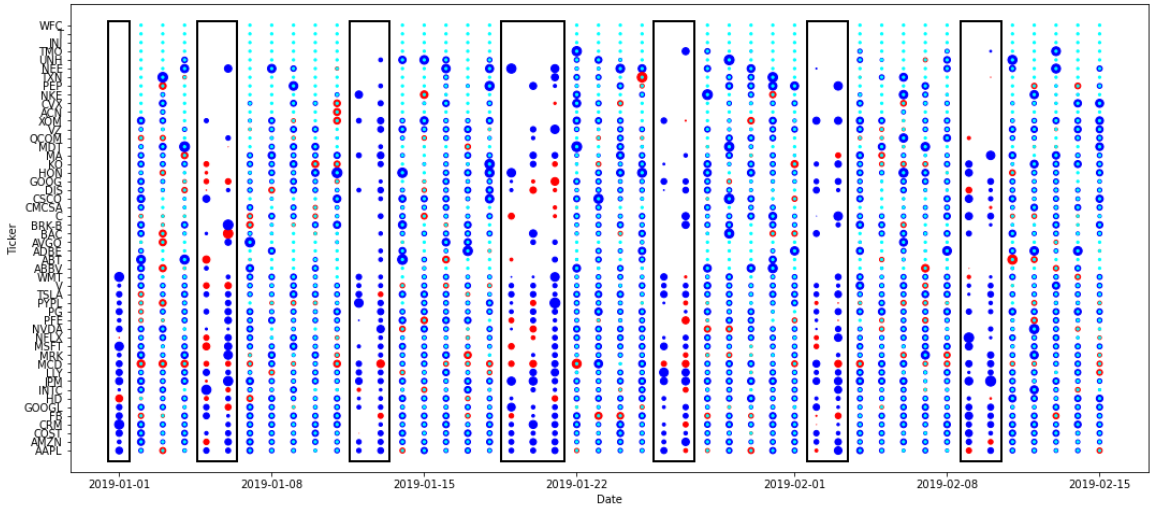
De forma visual en la Gráfica 3.4 no se identifican anomalías. Es posible observar que el precio y volumen de las transacciones tienen un comportamiento continuo, lo que indica que no hay datos faltantes; por otra parte, no se visualizan cambios abruptos que sean indicio de datos erróneos. Con ello se confirma que los datos se encuentran en orden para el análisis; en este caso se muestra la gráfica de una sola acción (*AAPL*) acotada a un periodo específico (45 días); sin embargo, es importante mencionar que el proceso de exploración y validación de los datos se realiza para todas las acciones y en el periodo de estudio (2014 a 2020). La gráfica de las diez primeras acciones del índice S&P500 de acuerdo con su orden alfabético se encuentran en la sección 6.3.



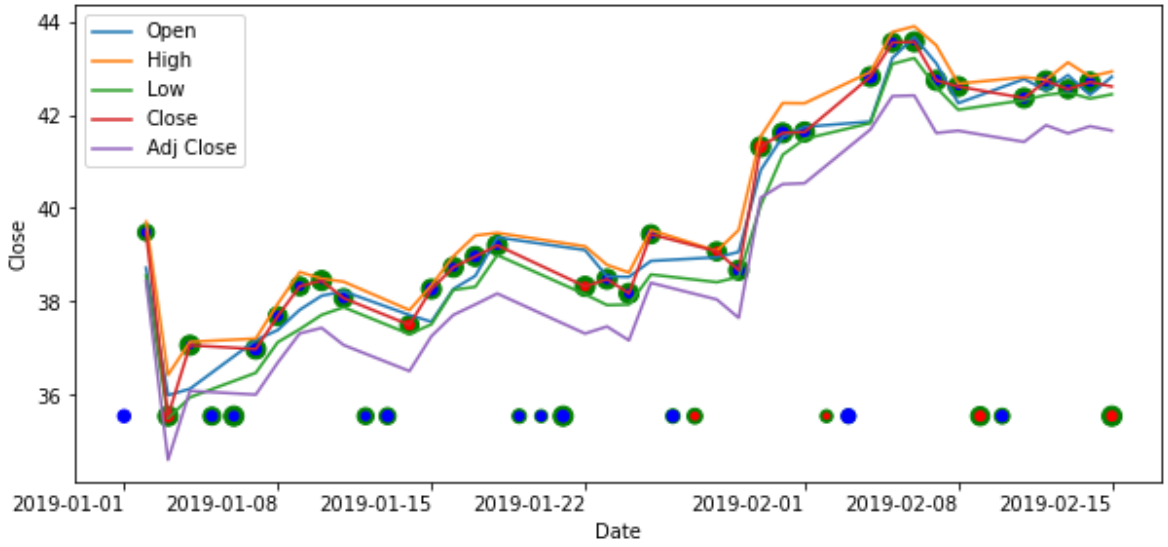
Gráfica 3.4 Gráfico de los atributos de la acción AAPL del conjunto de datos obtenido de Yahoo Finance. (a) Datos referentes al precio de la acción. (b) Datos del volumen de transacciones. Elaboración propia

3.3.3 Exploración integral de los datos recopilados

Por último, se revisa la distribución de los datos de forma conjunta. En la Gráfica 3.5 se muestra la distribución de los datos de las noticias financieras y de los datos transaccionales de cada acción; como se aprecia, existen periodos en los que no existen datos transaccionales. En esta gráfica, se hace evidente que estos periodos son días inhábiles para el mercado bursátil, tal como se identificó al revisar los datos de forma independiente; a partir de ello podemos tomar como cierta la aseveración. Adicional a ello, se hace una revisión de la interacción entre ambos conjuntos de datos; en la Gráfica 3.6 se integran los datos de las noticias financieras; y se visualizan otros atributos, tales como: el precio de cierre *Close* de la acción junto con el sentimiento de las noticias de forma diaria. En este caso, de forma generalizada se aprecia que los periodos en lo que hay un incremento en el precio de la acción, las noticias son catalogadas como positivas; y sucede de forma contraria cuando las noticas son negativas. Es importante notar que también existen periodos en los que a pesar de que las noticias son positivas, el precio de la acción baja. A pesar de que el propósito del presente trabajo no es dar explicación a este fenómeno, el modelo de aprendizaje automático propuesto sí tiene el propósito de hacer predicciones aun teniendo estas diferencias; en este sentido, se espera que la integración de otros datos como índices económicos e indicadores derivados del análisis bursátil técnico ayuden a que el modelo tenga una mayor precisión.



Gráfica 3.5. Diagrama de dispersión del sentimiento de cada noticia por día, en conjunto con la distribución de los datos transaccionales de cada acción. En rojo se señalan las noticias negativas, en azul oscuro las positivas; y en azul claro la existencia de datos transaccionales. Con fines ejemplificativos, solo se grafican los datos de las 25 acciones más representativas en un periodo de 45 días. Elaboración propia.



Gráfica 3.6. Gráfico de los atributos referentes a la acción AAPL en conjunto con los datos del sentimiento de cada noticia por día; en rojo se señalan las noticias negativas, y en azul las positivas; el tamaño de cada punto representa el grado de positividad o negatividad. Con fines ejemplificativos, solo se grafican los datos de las 25 acciones más representativas en un periodo de 45 días. Elaboración propia.

3.3.4 Preparación de los datos

Enseguida se calculan atributos derivados basados en el análisis técnico bursátil que se fundamenta en el supuesto de que los movimientos en el mercado son patrones que se repiten de forma cíclica. En este caso los atributos calculados sirven como indicadores de un movimiento positivo o negativo en el precio de las acciones; es importante mencionar que dichos indicadores son una referencia únicamente basada en el comportamiento del precio y volumen de transacciones anteriores. El indicador OBV (On-Balance Volume) refleja el sentimiento de los inversionistas por comprar o vender una acción.

$$\boldsymbol{OBV = OBV_{prev} + \begin{cases} Volumen; & Close > Close_{prev} \\ 0 & Close = Close \\ -Volumen & Close < Close_{prev} \end{cases}} \tag{3.1}$$

El OBV adiciona el volumen de transacciones cuando el precio de la acción sube, y lo sustrae cuando baja como se muestra en la ecuación (3.1) (Tsang & Chong, 2009); esta condición se implementa en Python con el código que se muestra a continuación.

```
#On-Balance-Volume (OBV)
   data[i]['OBV_temp']=0
   data[i]['OBV_temp']=np.where(data[i]['Close']>data[i]['Close'].shif(1),data[i]['Volume'],
                        np.where(data[i]['Close']==data[i]['Close'].shift(1),0,
                              -data[i]['Volume']))

   data[i]['OBV_temp'][0]=0
   data[i]['OBV']=(data[i]['OBV_temp']+ data[i]['OBV_temp'].shift(1)).cumsum()

   del data[i]['OBV_temp'] #Borrar columna temporal 'OBV_temp'
```

El indicador MACD (Moving Average Convergence Divergence) revela cambios en la dirección y duración de una tendencia en el precio de una acción. El indicador se compone por dos atributos: *MACD y MACD_Signal*, que en conjunto ayudan a determinar si, con base en datos previos, el precio de la acción presenta una tendencia a subir o bajar (Fernando, 2021). En las ecuaciones (3.2) y (3.3) (Mateu Gordon, s.f.) se muestra el método para obtener el indicador; sus dos atributos son resultado del promedio móvil exponencial (EMA); es decir la media de los datos durante el periodo de tiempo definido ponderada exponencialmente. El MACD es un indicador gráfico de líneas, el cruce entre ellas se interpreta como una señal que anticipa el cambio en la tendencia en el precio de la acción; el cálculo para el conjunto de datos que se analiza en este proyecto se muestra a continuación:

$$\boldsymbol{MACD = EMA(12) - EMA(26)} \tag{3.2}$$

$$\boldsymbol{MACD\_Signal = EMA(9, MACD)} \tag{3.3}$$

A continuación, se muestra el código en Python para el cálculo de los indicadores gráficos *MACD* y *MACD_Signal:*

```
#Exponential Moving Averages [EMA]
#alpha = [2/(period+1)]
data[i]['EMA_26'] = data[i]['Close'].ewm(alpha=0.074074074, adjust=False).mean()
data[i]['EMA_12'] = data[i]['Close'].ewm(alpha=0.153846154, adjust=False).mean()
data[i]['MACD'] = data[i]['EMA_12']-data[i]['EMA_26']
data[i]['MACD_Signal'] = data[i]['MACD'].ewm(alpha=0.2, adjust=False).mean()
```

El oscilador estocástico es un indicador porcentual de la sobrecompra o sobreventa de una acción; oscila entre cero y cien; y ayuda medir el precio actual de una acción en comparación con el rango de precio durante un periodo definido. Cuando su valor se encuentra por encima del 80% se considera que la acción está sobrecomprada, por el contrario, cuando se encuentra por debajo del 20% se infiere que la acción esta sobrevendida (Heyes, 2020). De forma similar al indicador MACD, el oscilador estocástico es un indicador gráfico compuesto por dos líneas que en conjunto ayudan a identificar tendencias a futuro. La ecuación (3.4) (Hayes, 2021) muestra el cálculo de la

línea (%K) considerando un periodo de 14 días. La ecuación (3.5) (Hayes, 2021) muestra el cálculo de la segunda línea (%D), la media móvil simple (SMA) de %K en un periodo de 3 días.

$$\%\boldsymbol{K} = \left(\frac{\boldsymbol{Close - Low(14)}}{\boldsymbol{High(14) - Low(14)}}\right)\boldsymbol{x\ 100} \tag{3.4}$$

$$\%\boldsymbol{D} = \boldsymbol{SMA}(\%\boldsymbol{K}, \boldsymbol{3}) \tag{3.5}$$

El código para su implementación en Python se muestra enseguida, mismo que, con referencia en el análisis técnico bursátil, y con base en la interacción entre los atributos *MACD, MACD_Signal*, *%K* y *%D* calcula nuevos valores que sirven como indicadores para cambios en el comportamiento de las acciones.

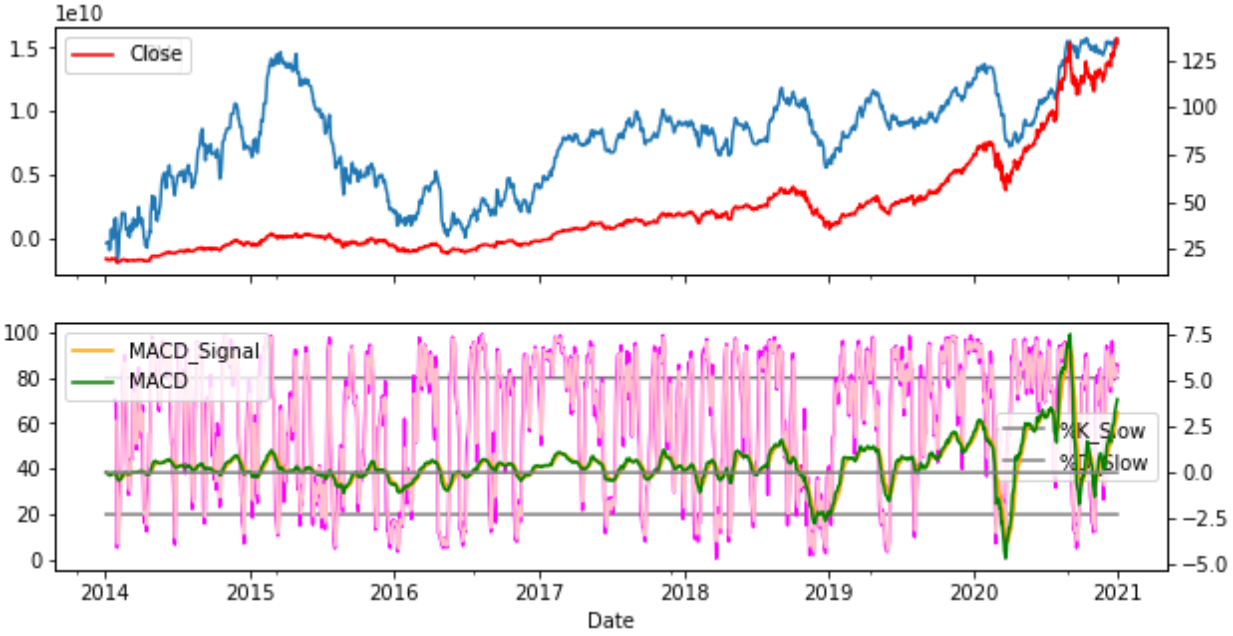
```
#Stochaist Oscillator
#https://pythonforfinance.net/2017/10/10/stochastic-oscillator-trading-strategy-backtest-in-python/
data[i]['L14'] = data[i]['Low'].rolling(window=14).min()
data[i]['H14'] = data[i]['High'].rolling(window=14).max()
data[i]['%K'] = 100*((data[i]['Close'] - data[i]['L14']) / (data[i]['H14'] - data[i]['L14']) )
data[i]['%K_Slow'] = data[i]['%K'].rolling(window=3).mean()
 data[i]['%D_Slow'] = data[i]['%K_Slow'].rolling(window=3).mean()
```

Por último, se calcula el porcentaje de variación diaria en el precio de cada acción, con el fin de identificar cambios significativos, y de esa forma encontrar correlaciones con otros atributos que sirvan para tener una mejor comprensión del comportamiento del mercado bursátil. El código en Python para obtener el atributo se muestra a continuación:

```
data[i]['var']=(data[i]['Close'].shift(1)-data[i]['Close'])/data[i]['Close'].shift(1)
```

La Gráfica 3.7 muestra los atributos calculados de la acción *AAPL;* en ella se observa el comportamiento de los atributos OBV, MACD, MACD_Signal, %K y %D durante el periodo de estudio. La interpretación del comportamiento de los atributos y la interacción entre ellos es de utilidad, entre otras cosas, para identificar condiciones y tendencias en el mercado en torno a una acción determinada. Para ello, se calculan los atributos *MACD crossover*, *Stochastic indicator*, y *MACD Cross* que categorizan los atributos numéricos *MACD*, *MACD_Signal*, *%K_Slow y*

%D_Slow con base en el rango en el que se encuentran, así como la interacción entre ellos, tal como se muestra en la Figura 3.6.



Gráfica 3.7 Gráfico de los atributos numéricos calculados para la acción AAPL. (a) *Close* y *OBV*; (b) *MACD_Signal*, *MACD*, *%K* y *%D*. Elaboración propia

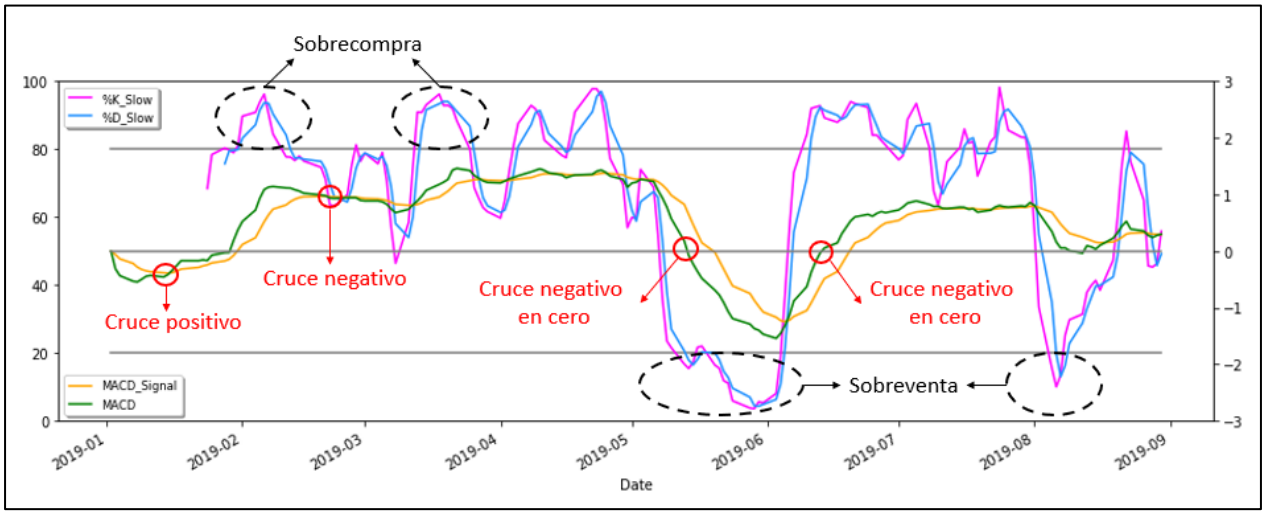


Figura 3.6. Representación de los atributos calculados con base en el análisis técnico. Elaboración propia con información de (StockCharts, s.f.) y (Hayes, 2021)

El código en Python para el cálculo se muestra a continuación:

```
data[i]['crossover?'] = np.where(data[i]['MACD'] <= data[i]['MACD_Signal'],
                        (np.where((data[i]['MACD'].shift(1) > 0) & (data[i]['MACD']<0) , 'Zero
Cross_Downwards',
                                  (np.where((data[i]['MACD']<0) , 'Down_Below_0', 'Down_Over_0'
                                  ))
                                  )),
                        (np.where((data[i]['MACD'].shift(1) < 0) & (data[i]['MACD']>0) , 'Zero Cross_Upwards',
                                  (np.where((data[i]['MACD']<0) , 'Up_Below_0', 'Up_Over_0'
                                  ))
                                  )))


   data[i]['Dif_MACD_Temp']=(data[i]['MACD']-data[i]['MACD_Signal'])+(data[i]['MACD'].shift(1)-
data[i]['MACD_Signal'].shift(1))
   data[i]['Dif_MACD_Temp-1']=data[i]['Dif_MACD_Temp'].shift(1)

   data[i]['Sign_MACD']= data[i]['Dif_MACD_Temp'].multiply(data[i]['Dif_MACD_Temp-1'])
   data[i]['Sign_MACD_cross']=np.where(data[i]['Sign_MACD']<0,
                              np.where(data[i]['Dif_MACD_Temp']<0,"C_Down","C_Up"),data[i]['crossover?'])
   del data[i]['Dif_MACD_Temp']
   del data[i]['Dif_MACD_Temp-1']
   del data[i]['Sign_MACD']

data[i]['s_indicator'] = np.where(((data[i]['%K_Slow'] <= data[i]['%D_Slow'])
                         & (data[i]['%K_Slow'].shift(1) > data[i]['%K_Slow'])
                        & (data[i]['%K_Slow'].shift(1) > 80)),
                        'Sell',
                        (np.where(((data[i]['%K_Slow'] >= data[i]['%D_Slow'])
                                 & (data[i]['%K_Slow'].shift(1) < data[i]['%K_Slow'])
                                 & (data[i]['%K_Slow'].shift(1) < 20)),
                                'Buy',
                                (np.where(data[i]['%K_Slow']>=80,"Overbougth",
                                (np.where(data[i]['%K_Slow']<=20,"Oversold","Hold")))))))
```

La Tabla 3-9 indica la representación de cada uno de los valores de los atributos *MACD crossover* y *Stochastic indicator*; como se observa, se describe el cruce de los atributos entre ellos, y en cero. El indicador será utilizado más adelante para identificar tendencias en el mercado de valores.

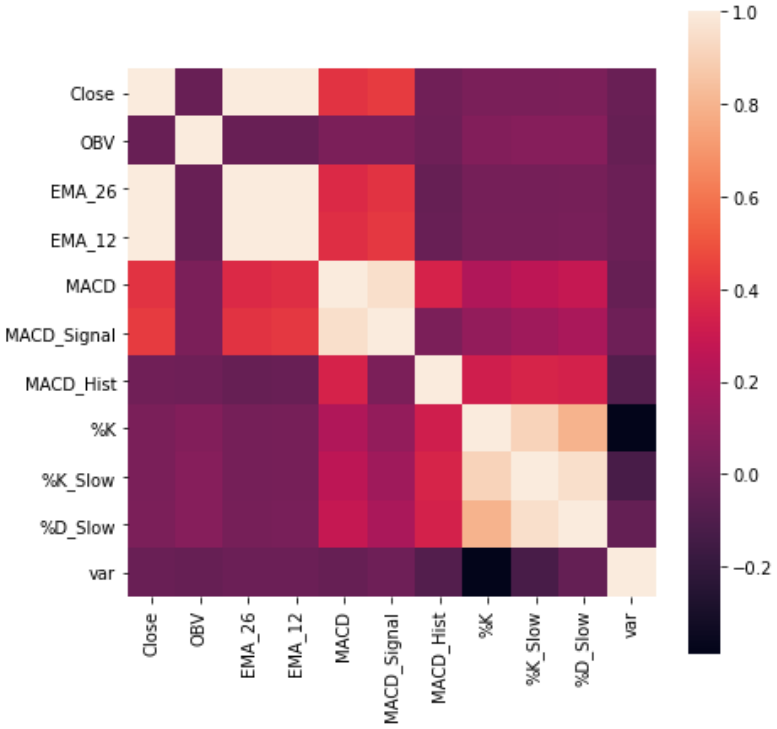
Tabla 3-9

Representación de los atributos MACD crossover y Stochastic

Atributo	Indicador	Representación
MACD crossover	C_Down	Cruce negativo de *MACD* sobre *MACD_Signal*
	C_Up	Cruce positivo de *MACD* sobre *MACD_Signal*
	Down_Below_0	*MACD* con tendencia negativa, por debajo de cero
	Down_Over_0	*MACD* con tendencia negativa por encima de cero
	Up_Below_0	*MACD* con tendencia positiva por debajo de cero
	Up_Over_0	*MACD* con tendencia positiva por encima de cero
	Zero Cross_Downwards	Cruce de *MACD* en cero hacia abajo
	Zero Cross_Upwards	Cruce de *MACD* en cero hacia arriba
Stochastic indicator	Sell	Valor de %K por encima de 80
	Buy	Valor de %K por debajo de 20
	Hold	Valor de %K entre 20 y 80

Nota: Representación basada en la publicación de StockCharts (StockCharts, s.f.)

Los atributos previamente calculados se estandarizan para hacer una revisión de la correlación entre ellos; la Gráfica 3.8 muestra que el precio de cierre *Close* tiene una correlación positiva con los atributos *MACD* (0.41) y *MACD_Signal (0.43);* también se aprecia una correlación negativa entre *%K* y el porcentaje de variación diaria en el precio de las acciones *var* (-0.38). Los atributos *MACD, MACD_Signal* y *%K* fueron calculados en función del precio de cierre de cada acción. En este sentido, la correlación hallada respalda la aseveración planteada por el análisis técnico, el cual indica que las tendencias en el mercado se pueden representar a partir de modelos matemáticos que relacionan los datos históricos del instrumento financiero; en este caso, el precio de cierre *Close*. Este hallazgo resulta relevante para el estudio, pues dichos atributos reflejan el comportamiento del mercado durante un periodo definido.



Gráfica 3.8 Correlación entre el precio de cierre (*Close);* atributos calculados (*OBV*, *EMA_26*, *EMA_12*, *MACD*, *MACD_Signal*, *MACD_Hist*, *%K*, *%K_Slow*, *%D_Slow);* y el porcentaje de variación diaria en el precio de las acciones (*% Variación).* Elaboración propia.

	Close	OBV	EMA_26	EMA_12	MACD	MACD_Signal	MACD_Hist	%K	%K_Slow	%D_Slow	var
Close	1.000000										
OBV	-0.017306*	1.000000									
EMA_26	0.998898*	-0.019588*	1.000000								
EMA_12	0.999496*	-0.018341*	0.999781*	1.000000							
MACD	0.408259*	0.047845*	0.372427*	0.391746*	1.000000						
MACD_Signal	0.431364*	0.049612*	0.403862*	0.421867*	0.953670*	1.000000					
MACD_Hist	0.009169*	0.004013	-0.023966*	-0.016018*	0.343722*	0.045274*	1.000000				
%K	0.041262*	0.072231*	0.024467*	0.029101*	0.215058*	0.123934*	0.327248*	1.000000			
%K_Slow	0.044358*	0.078303*	0.027814*	0.033389*	0.258132*	0.162037*	0.351344*	0.910439*	1.000000		
%D_Slow	0.045537*	0.080908*	0.029973*	0.036111*	0.283955*	0.193685*	0.338306*	0.795713*	0.956071*	1.000000	
var	-0.014371*	-0.022433*	-0.005181	-0.005672	-0.023784*	0.003036	-0.088450*	-0.388755*	-0.126707*	-0.030486*	1.000000

Figura 3.7. Tabla de correlaciones entre precio de cierre de las acciones (*Close*), el porcentaje de variación diaria en el precio de cierre (*%Variación*), y los atributos calculados. Los valores marcados con un asterisco indican que son significativamente estadísticos. En rojo se señalan las correlaciones cuyo valor es superior a 0.30 Elaboración propia

A partir del conjunto de datos inicial y de los atributos previamente calculados, se genera un nuevo conjunto de datos solo con los atributos que se utilizarán en las siguientes fases de este proyecto, quedando como se muestra en la Tabla 3-10. A pesar de que no se identifica una correlación elevada entre el atributo OBV y el resto de las variables, se mantiene en el conjunto de datos ya que su correlación con otros atributos es significativamente estadística.

Tabla 3-10

Conjunto de datos transaccionales de las acciones

Fecha	Ticker	Close	OBV	EMA 26	EMA 12	MACD	MACD Signal	MACD crossover	%K	Stochastic indicator	MACD Cross	% Variación
22/01/2014	AAPL	19.70	6.85E+08	19.57	19.54	-0.033	-0.067	Up_Below	71.34	Hold	Up_Below_0	-0.004
23/01/2014	AAPL	19.86	1.47E+08	19.60	19.59	-0.005	-0.055	Up_Below	86.74	Hold	Up_Below_0	-0.008
24/01/2014	AAPL	19.50	1.44E+08	19.59	19.58	-0.012	-0.046	Up_Below	53.40	Hold	Up_Below_0	0.018
27/01/2014	AAPL	19.66	1.57E+09	19.59	19.59	-0.004	-0.038	Up_Below	68.01	Hold	Up_Below_0	-0.008

Nota: Se muestran algunos registros de forma ejemplificativa. Datos no estandarizados. Elaboración propia.

Respecto a lo datos de los indicadores económicos de la Tabla 3-3; al ser la fuente de los datos *Intercontinental Exchange* (ICE), a través de *Yahoo Finance*, se asume que los datos se encuentran en orden. En este caso, no es necesario transformar los datos, ni calcular atributos derivados ya que por sí mismos representan el comportamiento de los indicadores económicos que se estudian en este proyecto. Como ejemplo, los datos que se muestran en la Tabla 3-11 corresponden a la volatilidad en el mercado de valores (^*VIX*) en cuyo atributo *Close* se aprecia que del 28 de enero al 31 de enero de 2014 la volatilidad aumentó; sin embargo, del 05 al 07 de febrero, la volatilidad disminuyó.

Tabla 3-11

Conjunto de datos de indicadores económicos

Fecha	Ticker	Close
28/01/2014	^VIX	15.80
29/01/2014	^VIX	17.35
30/01/2014	^VIX	17.29
31/01/2014	^VIX	18.41
…	…	…
05/02/2014	^VIX	19.95
06/02/2014	^VIX	17.23
07/02/2014	^VIX	15.29

Nota: Se muestran algunos registros de forma ejemplificativa. Datos no estandarizados. Elaboración propia.

3.3.5 Carga de datos en la base de datos multidimensional

Una vez que se han explorado y preparado los datos; se diseña el modelo de la base de datos multidimensional. Como se muestra en la Figura 3.8 la base de datos se conforma de tres dimensiones (*Datos Transaccionales, Sentimientos, Índices*), y una tabla de hechos (*Posición)*. Ya que el objetivo es establecer una relación entre los tres conjuntos de datos para determinar una postura de inversión para cada una de las acciones, la tabla *Posición* integra los datos de las dimensiones.

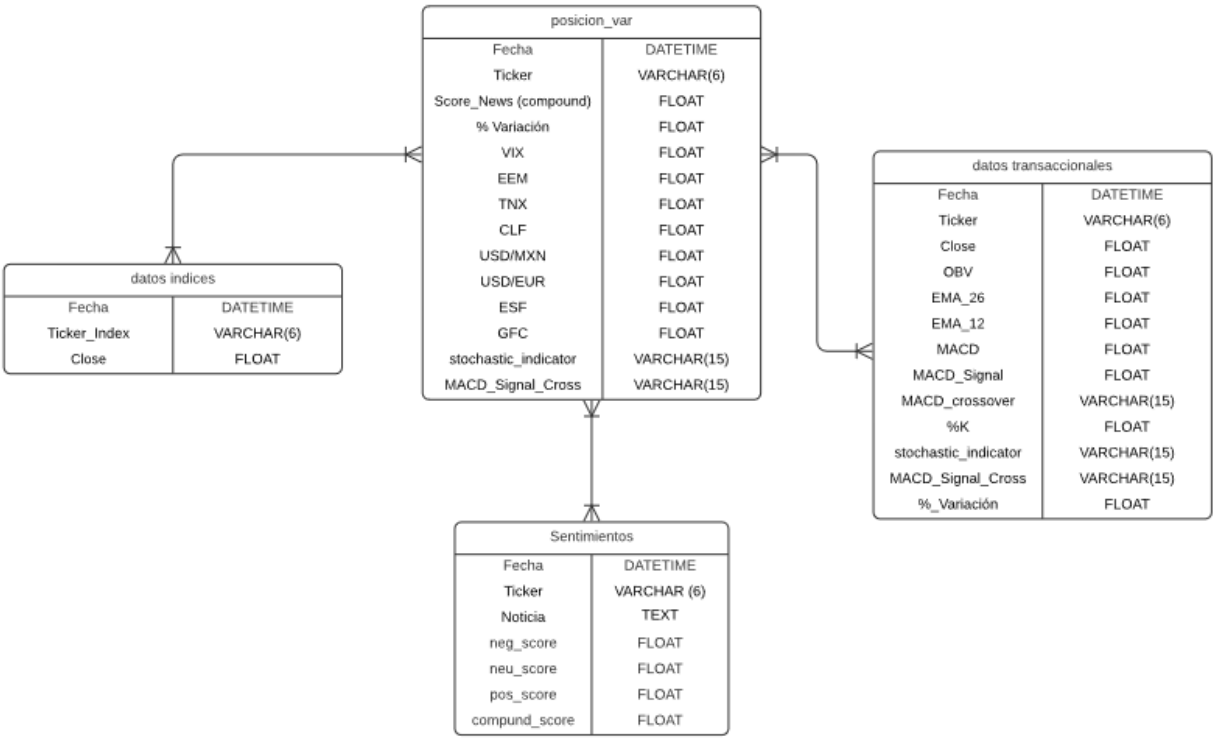


Figura 3.8. Modelo de la base de datos multidimensional. Elaboración propia.

El proceso para generar la base de datos, en la que se crearán las tablas y se cargarán los datos, se lleva a cabo utilizando el sistema manejador de bases de datos (SMBD) *MySQL Workbench 8.0* a través de *Jupyter Notebooks* con comandos disponibles en la librería *mysql.connector* de *Python,* el código para ello se muestra a continuación.

```
import mysql.connector
config = { 'user': 'root',  'password': 'password', 'host': '127.0.0.1',  'database': 'multidimensional',
'raise_on_warnings': True}
#Crear tablas en base de datos
cnx = mysql.connector.connect(**config)
cursor = cnx.cursor()
message_off = "SET sql_notes = 0;"
message_on = "SET sql_notes = 1;"
create_db_table = "\
CREATE TABLE IF NOT EXISTS `Datos Transaccionales` (\
  `Fecha` DATETIME,\
  `Ticker` VARCHAR(6),\
  `Close` FLOAT,\
  `OBV` FLOAT,\
```

```
  `EMA_26` FLOAT,\
  `EMA_12` FLOAT,\
  `MACD` FLOAT,\
  `MACD_Signal` FLOAT,\
  `MACD_crossover` VARCHAR(20),\
  `%K` FLOAT,\
  `stochastic_indicator` VARCHAR(15),\
  `MACD_Signal_Cross` VARCHAR(20),\
  `%_Variación` FLOAT\
);\
"
create_db_table2 = "\
CREATE TABLE IF NOT EXISTS `Sentimientos` (\
  `Fecha` DATETIME,\
  `Ticker` VARCHAR (6),\
  `Noticia` TEXT,\
  `neg_score` FLOAT,\
  `neu_score` FLOAT,\
  `pos_score` FLOAT,\
  `compound_score` FLOAT\
);\
"
create_db_table3 = "\
CREATE TABLE IF NOT EXISTS `indices` (\
  `Fecha` DATETIME,\
  `Ticker` VARCHAR(6),\
  `Close` FLOAT\
);\
"
create_db_table4 = "\
CREATE TABLE IF NOT EXISTS`Posicion` (\
  `Fecha` DATETIME,\
  `Ticker` VARCHAR(6),\
  `Score_News (compound)` FLOAT,\
  `% Variación` FLOAT,\
  `VIX` FLOAT,\
  `EEM` FLOAT,\
  `TNX` FLOAT,\
  `CLF` FLOAT,\
  `USD/MXN` FLOAT,\
  `USD/EUR` FLOAT,\
  `ESF` FLOAT,\
  `GFC` FLOAT,\
  `stochastic_indicator` VARCHAR(15),\
  `MACD_Signal_Cross` VARCHAR(20)\
);\
"
cursor.execute(message_off)
cursor.execute(create_db_table)
cursor.execute(create_db_table2)
cursor.execute(create_db_table3)
cursor.execute(create_db_table4)
cursor.execute(message_on)
cnx.close()
```

Los conjuntos de datos previamente generados se cargan de forma directa a la base de datos en las dimensiones correspondientes; en este caso se utilizan las librerías *sqlalchemy*[9] y *pymysql*[10]:

```
#Insertar datos en base de datos multidimensional
from sqlalchemy import create_engine
import pymysql

engine = create_engine("mysql+pymysql://{user}:{pw}@localhost/{db}"
        .format(user="root",
        pw="password",
        db="multidimensional"))

# Insertar conjunto de datos (transaccionales) en MySQL
    df.to_sql('datos transaccionales', con = engine, if_exists = 'append', chunksize = 1000, index= False)

# Insertar conjunto de datos (noticias) en MySQL
    parsed_and_scored_news.to_sql('sentimientos', con = engine, if_exists = 'append', chunksize = 1000, index=
False)

# Insertar conjunto de datos (índices) en MySQL
   df_indicators.to_sql('indices', con = engine, if_exists = 'append', chunksize = 1000, index= True)
```

El conjunto de datos para la tabla de hechos se genera a partir de las dimensiones a través de una consulta en SQL utilizando los atributos *Fecha* y *Ticker* como llaves para relacionar los datos. La integración de los datos de las tres dimensiones se realiza mediante la creación de dos vistas en *MySQL*, y utilizando el comando *JOIN* que permite combinar registros con atributos coincidentes de dos tablas; en este sentido, se llevan a cabo dos uniones (JOIN) una entre las dimensiones *Datos Transaccionales* y *sentimientos*; y otra entre la primera unión y la dimensión de *índices* como se muestra en el siguiente código:

```
CREATE VIEW indices_extended as (
SELECT
Fecha,
case when Ticker = "^VIX" then `Close` end as VIX,
case when Ticker = "EEM" then `Close` end as EEM,
case when Ticker = "^TNX" then `Close` end as TNX,
case when Ticker = "CL=F" then `Close` end as 'CL=F',
case when Ticker = "ES=F" then `Close` end as 'ES=F',
case when Ticker = "MXN=X" then `Close` end as 'MXN=X',
case when Ticker = "EUR=X" then `Close` end as 'EUR=X',
case when Ticker = "GC=F" then `Close` end as 'GC=F'
from indices
);
```

[9] Conjunto de herramientas para manipulación de bases de datos SQL en Python (SQL Alchemy, 2021)
[10] Librería MySQL para Python, permite conexión y consultas a bases de datos (Matsubara, 2021)

```
CREATE VIEW indices_Close_pivot as (
SELECT
Fecha,
SUM(VIX) as 'VIX',
SUM(EEM) as 'EEM',
SUM(TNX) as 'TNX',
SUM(`CL=F`) as 'CLF',
SUM(`ES=F`) as 'ESF',
SUM(`MXN=X`) as 'MXN',
SUM(`EUR=X`) as 'EUR',
SUM(`GC=F`) as 'GCF'
from indices_extended
group by Fecha
);


#Insertar conjunto de datos (posición) en MySQL
cnx = mysql.connector.connect(**config)
cursor = cnx.cursor(buffered=True)

carga_tabla_hechos ="\
INSERT INTO `posicion` \
SELECT \
A.Fecha,\
A.Ticker,\
A.Score_News,\
A.`%_Variación`,\
`indices_close_pivot`.VIX,\
`indices_close_pivot`.EEM,\
`indices_close_pivot`.TNX,\
`indices_close_pivot`.CLF,\
`indices_close_pivot`.MXN,\
`indices_close_pivot`.EUR,\
`indices_close_pivot`.ESF,\
`indices_close_pivot`.GCF,\
A.stochastic_indicator,\
A.MACD_Signal_Cross \
FROM \
(SELECT \
`datos transaccionales`.Fecha,\
`datos transaccionales`.Ticker,\
SUM(compound_score) AS 'Score_News',\
`datos transaccionales`.`%_Variación`,\
`datos transaccionales`.`stochastic_indicator`,\
`datos transaccionales`.`MACD_Signal_Cross` \
FROM \
`datos transaccionales` JOIN `sentimientos` ON \
`datos transaccionales`.Fecha = `sentimientos`.Fecha AND \
`datos transaccionales`.Ticker = `sentimientos`.Ticker \
GROUP BY \
`datos transaccionales`.Fecha, `datos transaccionales`.Ticker \
ORDER BY \
`datos transaccionales`.Ticker ASC,\
```

```
`datos transaccionales`.Fecha ASC) AS A \
JOIN `indices_close_pivot` ON \
A.Fecha = `indices_close_pivot`.Fecha \
GROUP BY \
A.Fecha, A.Ticker \
ORDER BY \
A.Ticker ASC,\
A.Fecha ASC \
"
for i in cursor.execute(carga_tabla_hechos, multi=True):
   pass
cursor.close()
cnx.commit()
cnx.close()
```

Los datos de los atributos *score news* y *% Variación* se estandarizan de forma que tengan una distribución con una media de cero y una desviación estándar de uno. La estandarización a partir de la ecuación (3.6) (Moreno Iglesias, 2020) permite realizar una comparación de los datos independientemente de sus unidades de medida.

$$x_{std} = \frac{x - \mu}{\sigma} \qquad (3.6)\ ^{11}$$

El código de la estandarización de los atributos numéricos es el siguiente:

```
#Estandarización de los datos de los atributos numéricos.
#https://www.dataquest.io/blog/python-pandas-databases/

cnx = mysql.connector.connect(**config)
cursor = cnx.cursor()
sql="SELECT * FROM multidimensional_2.posicion"
df = pd.read_sql_query(sql, cnx)
cnx.close()
# Aplicar la estandarización utilizando las funciones mean() y .std() methods
def z_score(df):
   # copy the dataframe
   df_std = df.copy()
   # apply the z-score method
   #for column in df_std.columns:
   df_std['Score_News (compound)'] = (pd.to_numeric(df_std['Score_News (compound)'],errors='ignore') -
pd.to_numeric(df_std['Score_News (compound)'],errors='ignore').mean()) / pd.to_numeric(df_std['Score_News
(compound)'],errors='ignore').std()
   df_std['% Variación'] = (pd.to_numeric(df_std['% Variación'],errors='ignore') - pd.to_numeric(df_std['%
Variación'],errors='ignore').mean()) / pd.to_numeric(df_std['% Variación'],errors='ignore').std()
   return df_std

df_test_S = z_score(df)
```

[11] x se refiere al dato observado; μ corresponde a la media del conjunto de datos; y σ es la desviación estándar de los datos

En la Tabla 3-12 se presenta un ejemplo del conjunto de datos de la tabla de hechos *Posición* que integra los datos de las dimensiones *datos transaccionales, sentimientos* e *índices*, y en la Gráfica 3.9 muestra el comportamiento de los datos de los atributos estandarizados *Score_News* y *%Variación* correspondientes a las acciones de *AAPL* que para fines explicativos considera un periodo de 165 días en ella se aprecia que generalmente cuando el puntaje de las noticias diarias de la acción es negativo, el porcentaje de variación tiene un comportamiento análogo, y viceversa. El resto de las acciones estudiadas en este proyecto tienen un comportamiento similar. Cabe destacar que en días específicos los atributos se comportan de forma inversa; es decir, que a pesar de que las noticias hayan sido clasificadas como negativas, la variación en el precio de la acción es positiva. Este hallazgo conlleva a plantear la pregunta sobre si la clasificación de las noticias es adecuada, o si existen ciertas noticias cuyo efecto sobre el precio de las acciones sea nulo; también es importante destacar que la tabla de hechos solo contiene datos de las noticias para los días en qué existe registro de transacciones en la bolsa de valores; es decir, no se muestran datos de las noticias publicadas durante fines de semana o días festivos, y por lo tanto no se visualiza su efecto en los días laborales subsecuentes. Un análisis incluyendo esos datos se propone para trabajos futuros.

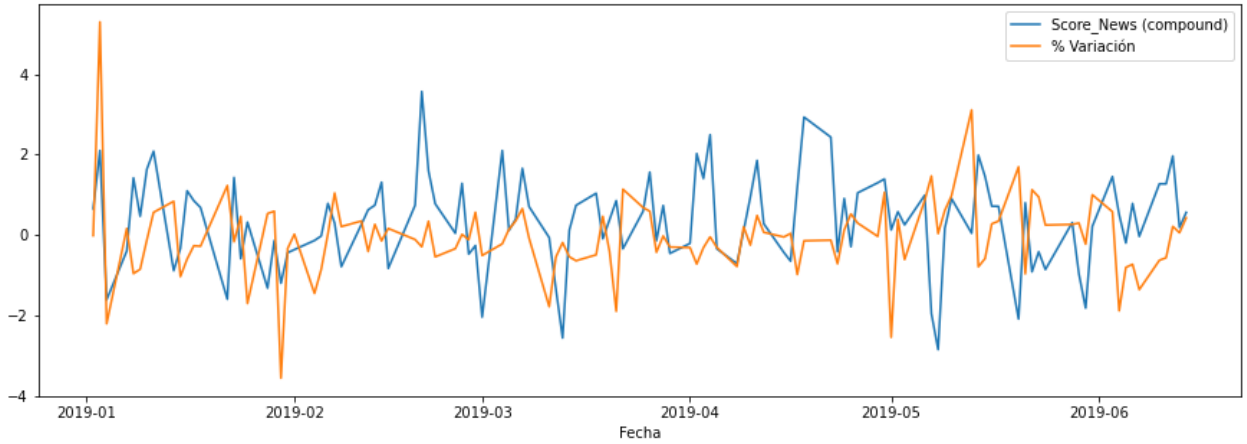
Haciendo una revisión de la distribución de los datos de la tabla de hechos es posible identificar su relación; la Gráfica 3.10 muestra una matriz de correlaciones en la que se identifica de forma gráfica que entre los indicadores económicos existen correlaciones del tipo lineal. En el caso de la relación entre *Score_News* y *% Variación,* como se muestra en la Gráfica 3.11, se aprecia una relación no lineal entre ellos; cuando el sentimiento de las noticias tiende hacia los extremos negativo y positivo, el porcentaje de variación se mueve a valores cercanos a cero; sin embrago, cuando el sentimiento tiende a la neutralidad, el porcentaje de variación se concentra alrededor de sus valores máximos y mínimos. En este sentido se plantea la hipótesis nula de que la variable *Score_News* se puede integrar al modelo predictivo en términos cuadráticos con el fin de tener una variable que represente la popularidad de la acción en los medios de comunicación independientemente del sentimiento; para ello se genera el atributo *Score_News_Sq*.

Tabla 3-12

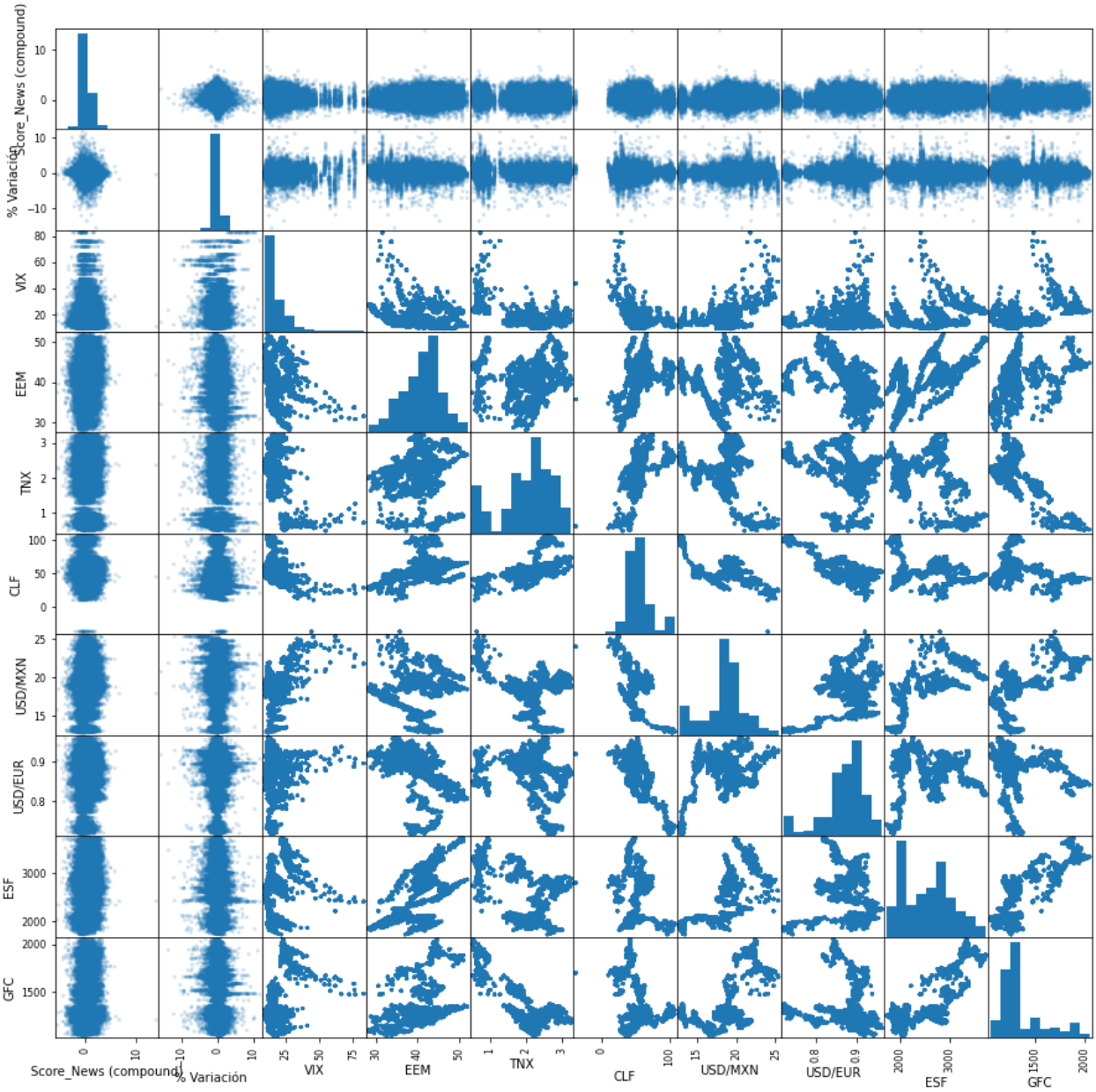
Conjunto de datos tabla de hechos "Posición"

Fecha	Ticker	Score_News (compound)	% Variación	VIX	EEM	TNX	CLF	USD MXN	USD EUR	ESF	GFC	Stochastic indicator	MACD_ Signal_Cross
03/01/2014	AAPL	0.6249	0.0220	13.76	40.12	2.995	93.96	13.094	0.732	1825.5	1238.4	Hold	Down_Below_0
06/01/2014	AAPL	2.2371	-0.0055	13.55	39.74	2.961	93.43	13.069	0.736	1820.75	1237.8	Hold	Down_Below_0
07/01/2014	AAPL	4.7702	0.0072	12.92	39.91	2.937	93.67	13.008	0.734	1830.75	1229.4	Hold	Down_Below_0
08/01/2014	AAPL	2.3469	-0.0063	12.87	39.78	2.993	92.33	13.125	0.734	1832.5	1225.3	Hold	Down_Below_0
09/01/2014	AAPL	-0.4253	0.0128	12.89	39.57	2.963	91.66	13.085	0.737	1833	1229.3	Hold	Down_Below_0

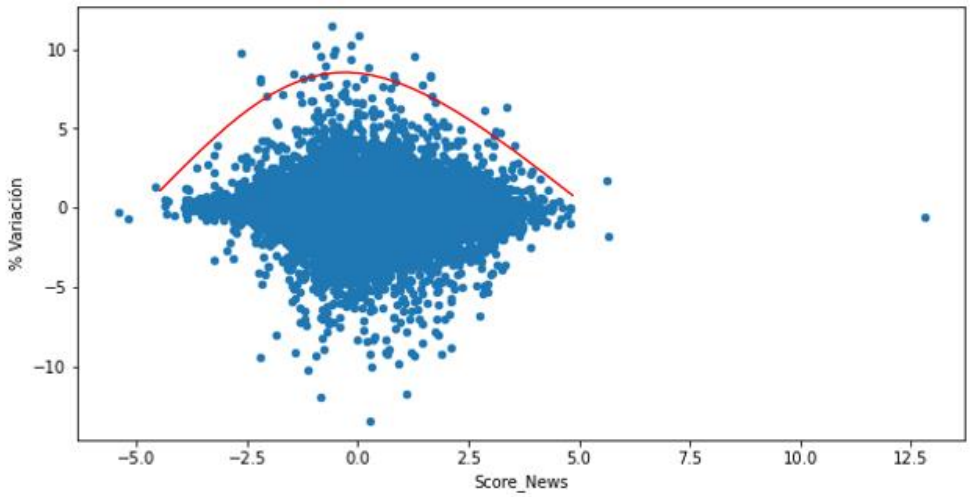
Nota: Se muestran algunos registros de forma ejemplificativa. Datos no estandarizados. Elaboración propia.



Gráfica 3.9 Comportamiento de los atributos numéricos de la tabla de hechos *Posición*. Se muestran datos de la acción *AAPL* durante un periodo definido de forma ejemplificativa. Elaboración propia.



Gráfica 3.10. Matriz de distribución de los datos de la tabla de hechos *Posición*. Elaboración propia.



Gráfica 3.11 Distribución de los datos del atributo *Score_News* con respecto al porcentaje de variación diaria de las acciones.

3.4 Modelado y evaluación

En esta fase de la metodología CRIS-DM se propone un modelo de aprendizaje supervisado para predecir la variación en el precio de las acciones estudiadas a partir de los datos de noticias financieras e índices económicos. Para ello, se analizan modelos con redes neuronales utilizando propagación simple y retropropagación como métodos para el entrenamiento, con el fin de definir cuál es el que mejores resultados ofrece. El modelo más simple de red neuronal es el perceptrón (Figura 3.9); se trata de una única neurona capaz de clasificar variables dicotómicas linealmente separables; y es la base para redes neuronales más complejas (Brownlee, 2020). Una red que mejora el comportamiento del perceptrón es la red *ADALINE* (Adaptive Linear Neuron) (Figura 3.10) desarrollada por Widrow & Hoff en los años sesenta; su algoritmo ajusta los parámetros de forma automática basada en datos de entrada previos (Norani, Shareduwan, & Kasihmuddin, 2021).

La principal diferencia entre el perceptrón y la red *ADALINE* es la función de activación, escalonada y lineal respectivamente; sin embargo, ambas son útiles principalmente para clasificar los datos en dos categorías. Un modelo de red neuronal multicapa con funciones de activación sigmoide tangencial y lineal, es capaz de hacer una regresión lineal; en este caso las variables pueden ser categóricas o numéricas

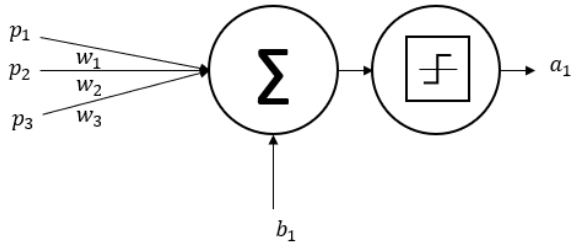


Figura 3.9 Perceptrón. Elaboración propia basado en (California State University Long Beach, s.f.)

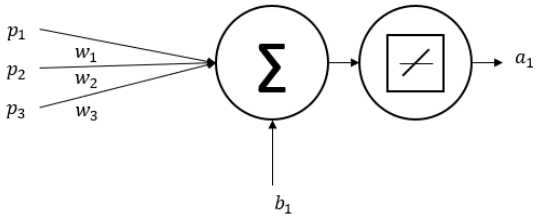


Figura 3.10 ADALINE Elaboración propia basado en (Kröse & Van der Smagt, 1996)

La Figura 3.11 ejemplifica un modelo de red neuronal multicapa; este recibe como patrones de entrada p_1a p_9 que corresponden a los datos numéricos de la tabla de hechos *Posición* del modelo multidimensional (incluyendo el atributo *Score_News_Sq*), y tiene como atributo de salida el porcentaje de variación diaria del precio de las acciones *% Variación*. En este caso la red neuronal multicapa se muestra con dos capas ocultas, cada una con 3 y 2 neuronas respectivamente. Sin embargo, en el presente trabajo se estudia el comportamiento de la red neuronal de forma heurística; es decir, a base de prueba y error mediante la variación de la cantidad de capas, de neuronas, y funciones de activación. En cada caso se evalúa el rendimiento del modelo con base en la eficiencia, y el error cuadrático medio (MSE). Valores bajos de MSE indican que el modelo se ajusta mejor a los datos (Harvill, 2020).

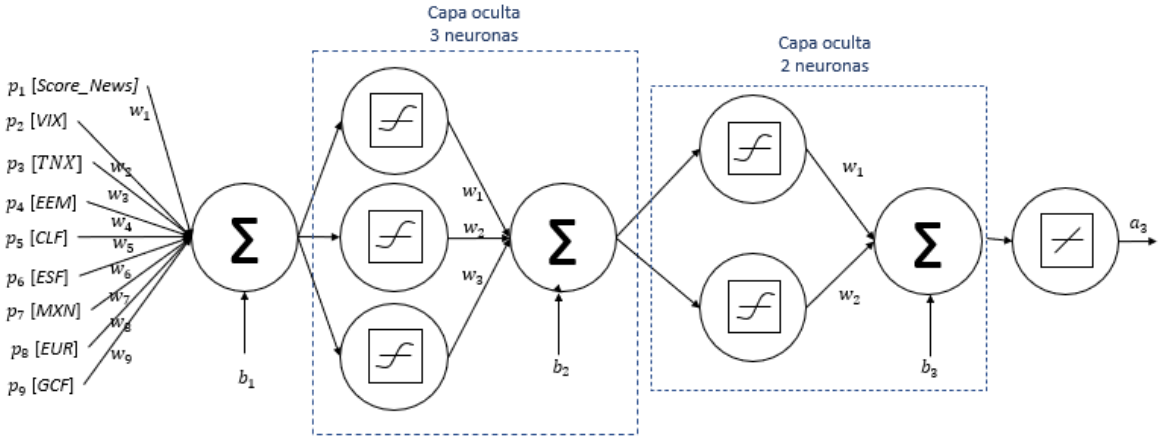


Figura 3.11. Ejemplo del modelo de red neuronal con dos capas ocultas, con 3 y 2 neuronas respectivamente. Elaboración propia.

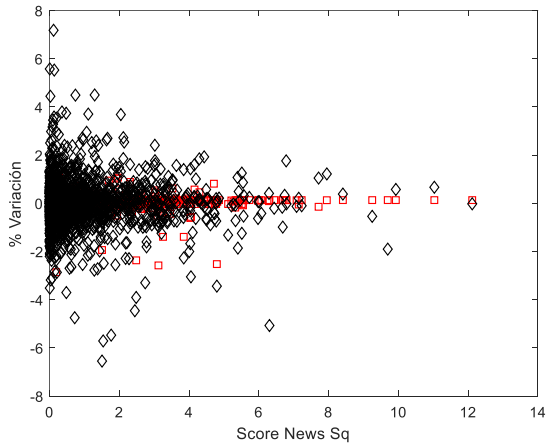
3.4.1 Modelo de regresión con redes neuronales

Para comenzar con el análisis, se plantea un modelo inicial de red multicapa con dos neuronas en una capa oculta; a partir de él, se agregan capas y neuronas a modo de mejorar el comportamiento predictivo del modelo. El algoritmo implementado en *Matlab* que se muestra a continuación es utilizado para modelar la red neuronal multicapa, realizar el aprendizaje automático con retropropagación, y evaluar la precisión del modelo mediante la estimación del error cuadrático medio (MSE). Para visualizar los resultados del modelo inicial, el código grafica la distribución de los datos originales y los obtenidos del modelo; así como el comportamiento MSE en cada una de las épocas.

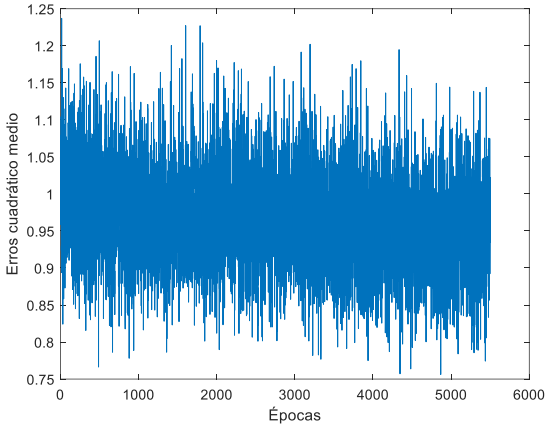
```
A = readmatrix('stocks.csv');
F = readtable('stocks.csv');
[nRows, nCols]=size(F);
datos_i =1;
cuenta = 1;
N=1;
for i=1:nRows
   if i>1
      cuenta = cuenta +1;
      if isequal(F(i-1,14),F(i,14))==0
         datos = cuenta;
         P = A(datos_i:datos,1:10)';
         T = A(datos_i:datos,11)';
         Q=size(P,2);
```

```
        [nRows, nCols]=size(P);
        J=length(P);
        n1=2; %Número de neuronas en la capa oculta
        n2=1;  %Número de neuronas en la capa oculta
        ep =0.75; %Ventana de valores iniciales
        %Valores iniciales de los pesos sinápticos
        W1=ep*(2*rand(n1,nRows)-1);
        b1=ep*(2*rand(n1,1)-1);
        W2=ep*(2*rand(n2,n1)-1);
        b2=ep*(2*rand(n2,1)-1);
        alpha = 0.00125;
        epocas = 5000;
        for iTe = 1:epocas
           suma=0;
           for j=1:J
              q=randi(Q);
              %Propagación de la entrada hacia la salida
              a1=tansig(W1*P(:,q)+b1); %salida de la primara capa
              a2(q)=(W2*a1+b2); %salida de la red
              %Retropropagación de la sensibilidad
              e=T(q)-a2(q);
              s2=-2*1*e;
              s1=diag(1-a1.^2)*W2'*s2;
              %Actualización de pesos sinápticos y polarizaciones
             W2=W2-alpha*s2*a1';
              b2=b2-alpha*s2;
              W1=W1-alpha*s1*P(:,q)';
              b1=b1-alpha*s1;
              %Sumando el error cuadrático
              suma = e^2+suma;
           end
           %Error cuadrático medio
           emedio(iTe)=suma/Q;
        end
        figure(1)
        plot(emedio)
        hold on;
        %Verificación de la respuesta multicapa
        for q =1:Q
           a1=tansig(W1*P(:,q)+b1); %salida de la primara capa
           a2(q)= (W2*a1+b2);
        end
        p=A(datos_i:datos,1:10)';
        for j=1:length(p)
          a1=tansig(W1*p(:,j)+b1);
          a_test(j)=tansig(W2*a1+b2);
        end
        figure(2)
        plot(p(3,:),a_test,'sr',A(datos_i:datos,3),T,'dk')
        hold on;
        datos_i=datos+1;
      end
   end
end
```

La Gráfica 3.12 muestra, en color negro, la distribución de los datos reales de los sentimientos de las noticias al cuadrado (Score_News_Sq) contra porcentaje de variación diaria *(% Variación*); y en color rojo los datos predichos. La Gráfica 3.13 muestra el error cuadrático medio por época tras 5,500 épocas. Ambas gráficas corresponden al modelo inicial propuesto con dos neuronas en una capa oculta. Como se puede ver, los datos estimados por el modelo se encuentran concentrados alrededor de un porcentaje de variación diaria cercana a cero; además, el error cuadrático medio, aunque parece tener una tendencia a la baja; se mantiene con valores elevados que oscilan entre 1.2 y 0.75; por esta razón, se concluye que el modelo no es adecuado para realizar la predicción. El error cuadrático medio del modelo tras cinco mil quinientas épocas es 1.0205; es decir que los datos estimados por el modelo se encuentran a una distancia media al cuadrado de 1.0205 de los datos reales.



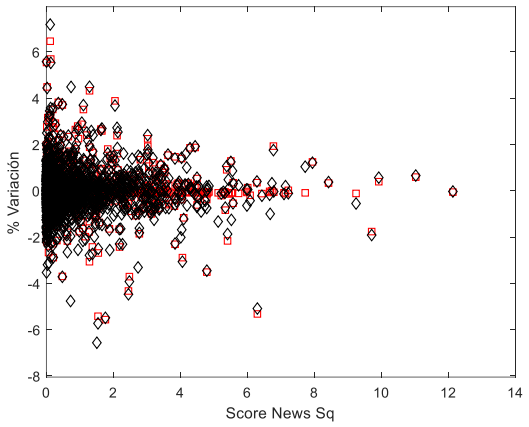
Gráfica 3.12. Distribución de los datos del atributo (Score_News_Sq) contra porcentaje de variación diaria (% Variación) del modelo inicial de la red neuronal. En color negro se muestran los datos reales, y en rojo los datos obtenidos de la red neuronal. Datos correspondientes a la acción AAPLE de 2014 a 2020.



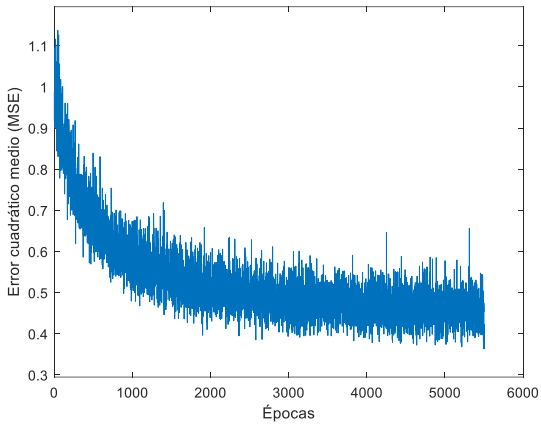
Gráfica 3.13. Error cuadrático medio por época del modelo inicial tras cinco mil quinientas épocas. Elaboración propia

Se revisan variaciones del modelo inicial, incrementando la cantidad de neuronas y de capas con el objetivo de reducir el error cuadrático medio, y mejorar su capacidad predictiva; incrementar el número de neuronas de dos a cinco en la capa oculta ayuda a que el MSE pase de 1.0205 a 0.8158. Así mismo, cuando en el modelo se consideran cincuenta neuronas en su única capa oculta

el MSE resultante es 0.4923. La raíz cuadrada del error cuadrático medio (RMSE) simplifica la interpretación al mostrar los valores en las mismas unidades que los datos del porcentaje de variación diaria (*% Variación*); en este caso corresponde a 0.71; es decir que el modelo genera predicciones del porcentaje de variación diaria cuyo error medio es 0.71%. A pesar de que el modelo con una capa oculta y cincuenta neuronas tiene un mejor desempeño en comparación con el modelo inicial; se revisa un modelo con dos capas ocultas, con cincuenta, y dos neuronas respectivamente; la Gráfica 3.14 y Gráfica 3.15 muestran el resultado. Se puede ver que los datos estimados tienen una distribución más cercana a los datos reales, a la vez que el error cuadrático medio presenta una tendencia a la baja significativa durante las primeras dos mil épocas, y posteriormente oscila entre los valores 0.6 y 0.4. En este modelo, tras cinco mil quinientas épocas el MSE es 0.4835; es decir 0.0088 menor al del modelo con una única capa oculta.



Gráfica 3.14. Distribución de los datos del atributo (*Score_News_Sq*) contra porcentaje de variación diaria *(% Variación*) del modelo con dos capas ocultas. En color negro se muestran los datos reales, y en rojo los datos obtenidos de la red neuronal. Datos correspondientes a la acción AAPLE de 2014 a 2020. Elaboración propia.



Gráfica 3.15. Error cuadrático medio por época del modelo con dos capas ocultas tras cinco mil quinientas épocas. Elaboración propia.

Considerando que la integración de neuronas y capas ocultas contribuyen positivamente a mejorar el comportamiento del modelo, se exploran diferentes combinaciones; los resultados de cada una de las variantes se muestran en la Tabla 3-13. Un modelo con tres capas ocultas y con

cincuenta, veinte, y doce neuronas respectivamente resulta ser el modelo óptimo pues el MSE resultante es 0.0310; es decir un valor de RMSE de 0.1760. %. La *Gráfica 3.16* y la *Gráfica 3.17* muestran los resultados del modelo; en ellas se puede ver que el error disminuye más rápidamente y de forma más suave, y que la distribución de los datos estimados por el modelo es más aproximada a la distribución de los datos reales; por esta razón, para los efectos de este proyecto se considera como el modelo más adecuado. Aunque este proyecto no considera el tiempo computacional como un factor para la selección del modelo más apropiado, se mide y se presenta de manera informativa únicamente para el modelo con el que se obtiene el porcentaje de precisión más elevado. El entrenamiento del modelo con tres capas ocultas, 50, 20 y 12 neuronas en cada una de ellas respectivamente, toma 6.63 minutos[12].

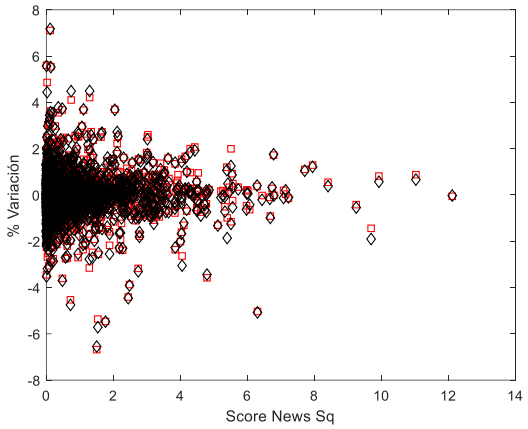
Tabla 3-13

Error cuadrático medio (MSE) de los modelos revisados

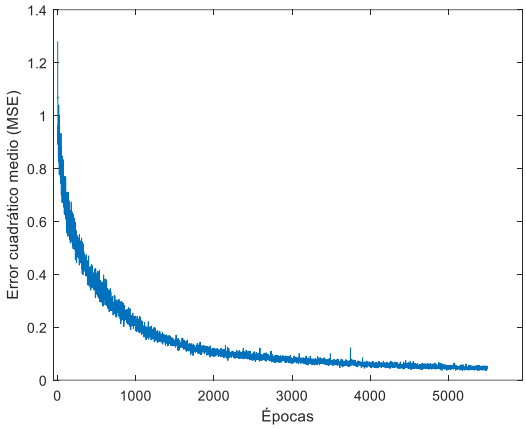
Capas ocultas	Neuronas por capa	Épocas	MSE
1	2	5,500	1.0205
1	5	5,500	0.8158
1	20	5,500	0.6756
1	50	5,500	0.4923
2	50,2	5,500	0.4835
2	50,10	5,500	0.1532
2	50,15	5,500	0.0962
2	50,20	5,500	0.0740
3	50,20,5	5,500	0.0927
3	50,20,12	5,500	0.0310

Nota: Se muestran el resumen de resultados de los modelos analizados. Datos no estandarizados. Elaboración propia.

[12] El entrenamiento realizado en un equipo con las siguientes características:
Procesador AMD Ryzen 3 3300U con Radeon Vega Mobile Gfx 2.10 GHz
Memoria RAM 12.0 GB (9.92 GB utilizable)

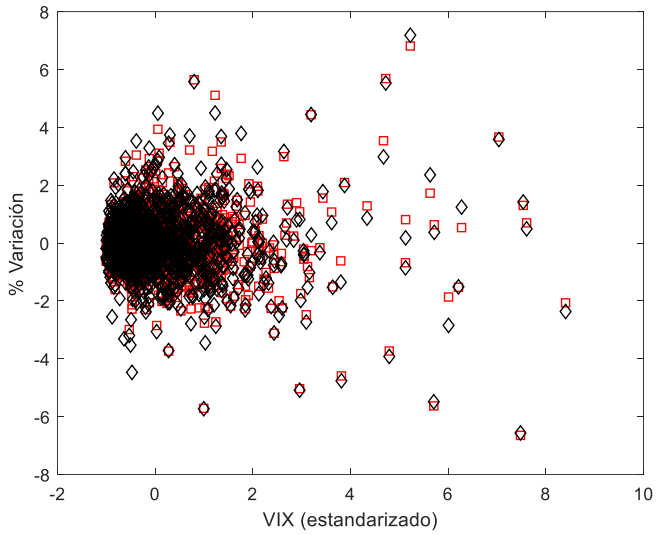


Gráfica 3.16. Distribución de los datos de los sentimientos de las noticias al cuadrado (Score_News_Sq) contra porcentaje de variación diaria (% Variación) de la red neuronal con tres capas ocultas. En color negro se muestran los datos reales, y en rojo los datos obtenidos de la red neuronal. Datos correspondientes a la acción AAPLE de 2014 a 2020. Elaboración propia.

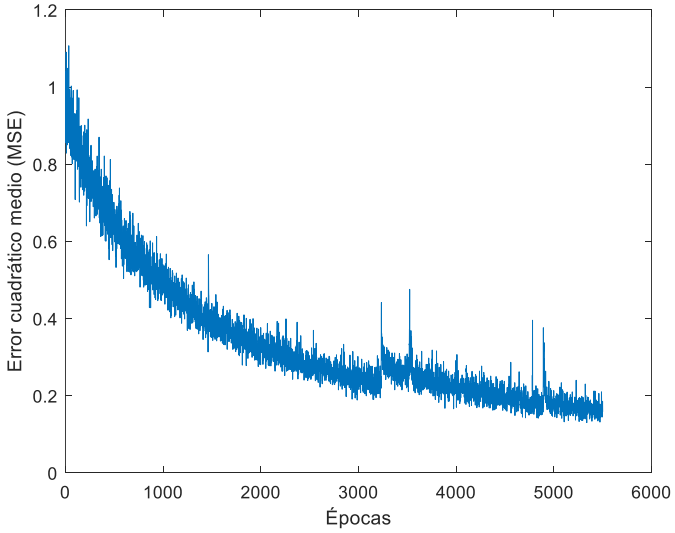


Gráfica 3.17. Error cuadrático medio por época del modelo con tres capas ocultas tras 5,500 épocas. Elaboración propia

Por último, tomando como base el último modelo de tres capas ocultas, se hace una revisión, pero esta vez sin integrar los datos de las noticias financieras; esto con el objetivo de identificar si las noticias juegan un papel relevante en los resultados de la predicción, y aceptar o rechazar la hipótesis nula planteada previamente sobre que el atributo *Score_News_Sq* contribuye a mejorar el comportamiento del modelo. Primero se revisa el comportamiento del modelo sin integrar los atributos *Score_News* y *Score_News_Sq* en el conjunto de datos, y utilizando los mismos parámetros, tras cinco mil quinientas épocas se obtiene un error cuadrático medio (MSE) de 0.158. La Gráfica 3.18 muestra los datos que arroja el primer modelo (en color rojo), que parecen tener un buen ajuste con los datos originales (color negro); sin embargo, como se aprecia en la Gráfica 3.19, el error es mayor en comparación con los que integra los datos de las noticias financieras; en este contexto es posible afirmar que las noticias financieras contribuyen a una mejora significativa en la predicción del modelo de redes neuronales propuesto.



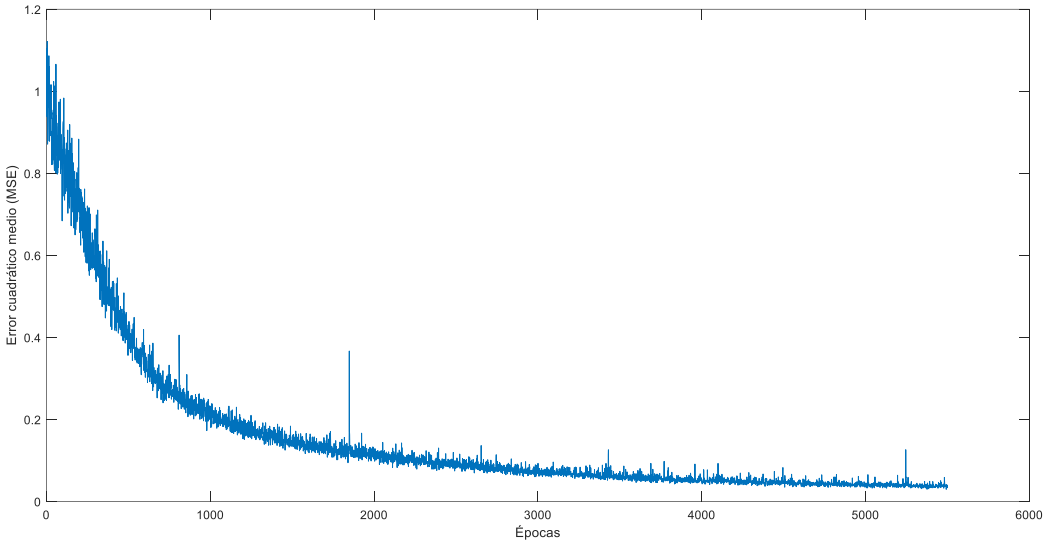
Gráfica 3.18. Distribución de los datos del porcentaje de variación diaria (% Variación) contra los datos de volatilidad (*VIX*) de la red neuronal con cinco capas ocultas. No se integran noticias financieras. En color negro se muestran los datos reales, y en rojo los datos obtenidos de la red neuronal. Datos correspondientes a la acción AAPLE de 2014 a 2020.



Gráfica 3.19. MSE por época del modelo con tres capas ocultas: 50,20,12 neuronas. No se integran noticias financieras. Elaboración propia

Ahora se revisa el modelo, pero sin integrar únicamente el atributo *Score_News_Sq* en el conjunto de datos. De igual forma se mantienen sin cambios los parámetros del modelo. Como se

puede ver en la Gráfica 3.20, el error cuadrático medio (MSE) presenta un comportamiento similar al del modelo en el que se integra la totalidad de los datos*;* sin embargo, en este caso el valor del error cuadrático medio es 0.0017 mayor (0.0327). Esta diferencia expresada como la raíz cuadrada del error cuadrático medio (RMSE) representa un 0.041% de la variación. El bajo error en el porcentaje de la variación de primera instancia parece no representar una diferencia significativa en la precisión del modelo; pero dado que el propósito de la predicción es que el inversionista pueda tomar decisiones que le lleven a incrementar su patrimonio, en este contexto se acepta la hipótesis nula y se mantiene el atributo *Score_News_Sq* en el conjunto de datos.



Gráfica 3.20. Error cuadrático medio del modelo de redes neuronales de tres capas (50, 20, 12 neuronas respectivamente) sin integrar el atributo Score_News_Sq.

La Tabla 3-14 muestra un resumen de los resultados de la evaluación del modelo, con 50,20 y 12 neuronas en cada una de sus capas ocultas, en los tres escenarios: integrando la totalidad de los atributos numéricos de la tabla de hechos *Posicion,* descartando los atributos *Score_News* y *Score_News_Sq.*

Tabla 3-14

Resumen de la evaluación del modelo sin integrar datos de noticias financieras

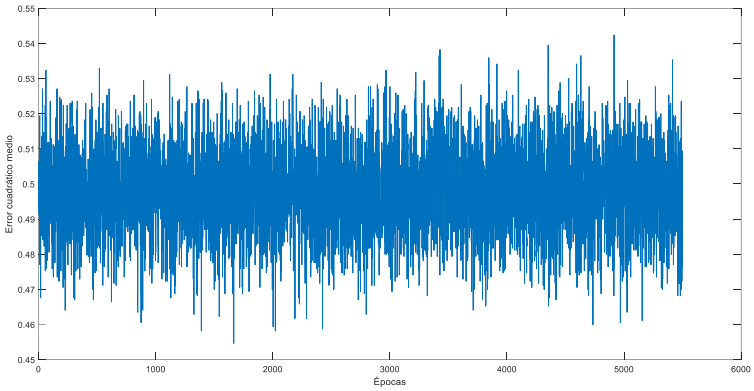
Atributos que se ignoran en el conjunto de datos	Error cuadrático medio (MSE)
Ninguno	0.0310
Score_News y *Score_News_Sq*	0.1580
Score_News_Sq	0.0327

Nota: Se muestran el resumen del error cuadrático medio (MSE) obtenido para el modelo de redes neuronales con tres capas integrando la totalidad de los datos, y descartando los datos de las noticias financieras. Elaboración propia.

El propósito de la predicción utilizando redes neuronales artificiales es predecir el porcentaje de la variación diaria en el precio de las acciones a partir de datos históricos; sin embargo, hasta ahora se ha revisado el modelo únicamente con datos en concordancia; es decir, se predice el porcentaje de variación del mismo día en que los datos de las noticias, transacciones e índices son generados. Esta forma de proceder, aunque proporciona resultados con un error cuadrático medio de 0.0310, no resulta útil pues algunos datos, por ejemplo, los de las noticias financieras se producen aún después del cierre de operaciones en el mercado de valores.

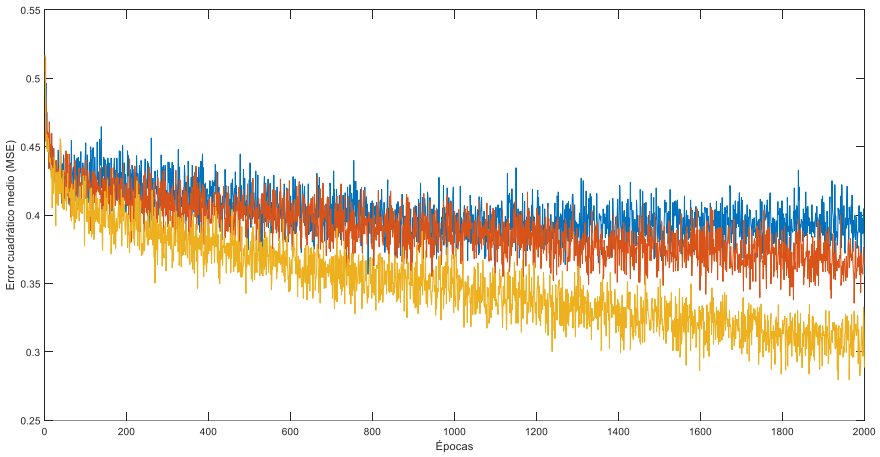
3.4.2 Modelo de clasificación con redes neuronales

Con la finalidad de mejorar la precisión del modelo, paralelamente se revisa un modelo de red neuronal para hacer una predicción del sentido de la variación diaria en el precio de las acciones; es decir, a partir de los datos de entrada determinar si la variación esperada para una fecha determinada será ~~es~~ positiva o negativa. Para el caso, se genera una nueva columna en el conjunto de datos cuyos valores para identificar una variación positiva y negativa son 1 y 0 respectivamente. De igual forma partimos de un modelo simple con una sola neurona. El modelo inicial no ofrece una solución pues el error medio en lugar de reducir durante cada época, se mantiene oscilante entre los valores 0.52 y 0.46. La *Gráfica 3.21* muestra el gráfico del error cuadrático medio (MSE) tras cinco mil épocas.



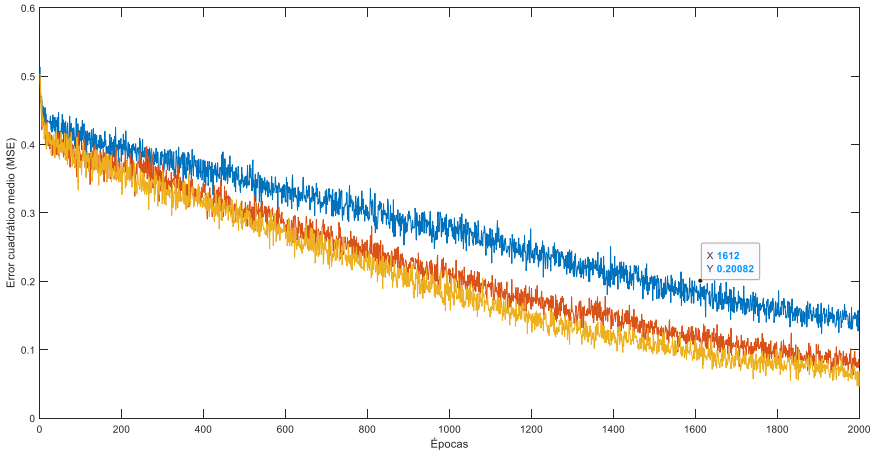
Gráfica 3.21. Error cuadrático medio del modelo de red neuronal con una única neurona para clasificación del sentido de la variación del precio. Elaboración propia.

Dado que con una única neurona no es posible realizar la clasificación, se explora un modelo con una capa oculta, al que se van añadiendo neuronas hasta lograr reducir el error de forma significativa. Como se aprecia en la Gráfica 3.22, a partir de ochenta neuronas, se comienza a percibir una tendencia a la baja en el error cuadrático medio, y una eficiencia en la clasificación del 62.42%. Aumentando el número de neuronas a cien, mejora el resultado; sin embargo, no de manera significativa pues la eficiencia incrementa menos del 1%. A partir de ciento veinte neuronas, el error disminuye notoriamente y la eficiencia del modelo aumenta a 71.2%.



Gráfica 3.22. Error cuadrático medio en modelo con una capa oculta. En color azul: ochenta neuronas, en naranja: cien neuronas, en amarillo: ciento veinte neuronas. Elaboración propia.

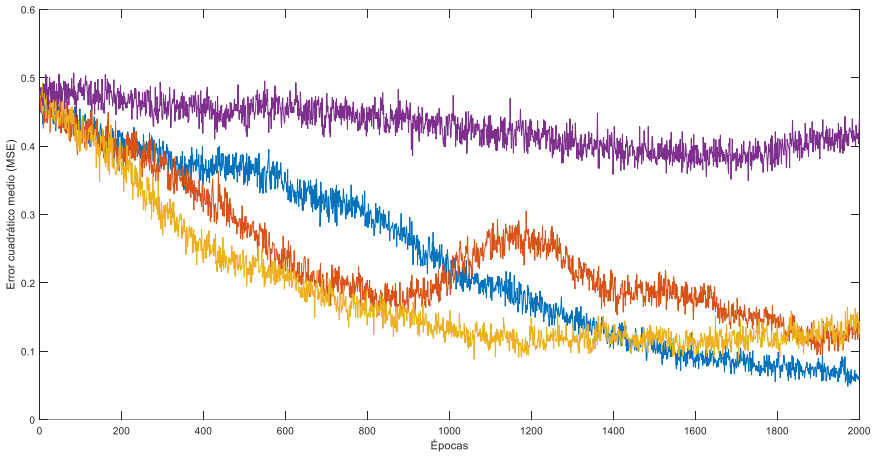
Manteniendo ciento veinte neuronas en la primera capa oculta, se revisa el comportamiento del modelo agregando una segunda capa. Primero se prueba el modelo con una neurona en su segunda capa y se va incrementando el número de neuronas; cuando el modelo tiene cincuenta neuronas en la segunda capa se obtienen mejoras significativas en comparación con el modelo con una sola capa oculta. El porcentaje de eficiencia del modelo con esa configuración es 80%; sin embargo, como se muestra en la Gráfica 3.23, aumentar la cantidad de neuronas a cien, reduce el error cuadrático medio (MSE) y consecuentemente mejora la eficiencia del modelo a 92.8%. Aún se mejora el rendimiento del modelo, con ciento veinte neuronas en ambas capas ocultas; con esa configuración la eficiencia alcanzada es del 94.59%



Gráfica 3.23. Comportamiento del error cuadrático medio (MSE) en el modelo con dos capas ocultas. En color azul: cincuenta neuronas en la segunda capa oculta; en color naranja: cien neuronas en la segunda capa oculta; en color amarillo: ciento veinte neuronas en la segunda capa ocultan Elaboración propia

Como siguiente paso se revisa un modelo con tres capas ocultas partiendo del modelo con ciento veinte neuronas en sus primeras dos capas. De igual forma se inicia con un número reducido de neuronas en la tercera capa oculta y se incrementa paulatinamente para comparar el rendimiento. Como se puede ver en la Gráfica 3.24, con cinco neuronas en la última capa, el rendimiento del modelo en lugar de mejorar, empeora; sin embargo, teniendo veinticinco neuronas en la última capa; el error cuadrático medio (MSE) se encuentra por debajo de 0.07; el modelo con esta configuración es capaz de clasificar correctamente el 96.4% de los datos, Por el contrario, al incrementar el número de neuronas a cincuenta y ciento veinte, en lugar de mejorar la precisión del

modelo, la empeora, pues su precisión baja a 84.86% y 75.91% respectivamente. El modelo de red neuronal con tres capas ocultas; ciento veinte neuronas en sus primeras dos capas y veinticinco neuronas en su última capa permite realizar la clasificación del sentido de la variación en el precio de las acciones con un nivel de precisión del 96.4%, por lo que sirve como complemento del modelo de red neuronal que realiza la regresión del porcentaje de variación diaria. En este caso, ejecutar ambos modelos tiene como consecuencia que el tiempo computacional sea mayor en comparación con la ejecución de uno solo.



Gráfica 3.24. Comportamiento del error cuadrático medio (MSE) en el modelo con tres capas ocultas. En color morado: cinco neuronas en la tercera capa oculta; en color azul: veinticinco neuronas en la tercera capa oculta; en color naranja: cincuenta neuronas en la segunda capa oculta; en color amarillo: cien neuronas en la tercera capa oculta. Elaboración propia.

La Tabla 3-15 muestra el resumen de los modelos revisados; en ella se observa que la precisión por encima del 60% se obtiene al utilizar ochenta neuronas con una capa oculta. Al considerar más neuronas y más capas ocultas, se logra obtener un porcentaje de precisión de hasta 96.40% con la desventaja de que el tiempo computacional para realizar el entrenamiento también incrementa. El entrenamiento del modelo con tres capas ocultas, 120, 120 y 25 neuronas en cada una de ellas respectivamente, toma 21.5 minutos[13].

[13] El entrenamiento realizado en un equipo con las siguientes características:
Procesador AMD Ryzen 3 3300U con Radeon Vega Mobile Gfx 2.10 GHz
Memoria RAM 12.0 GB (9.92 GB utilizable)

Tabla 3-15

Precisión de los modelos revisados

Capas	Neuronas por capa	Épocas	Precisión
1	80	2,000	64.42%
1	120	2,000	71.20%
2	120,50	2,000	80.00%
2	120,100	2,000	92.80%
2	120,120	2,000	94.59%
3	120,120,25	2,000	96.40%
3	120,120,50	2,000	84.86%
3	120,120,120	2,000	75.9%

Nota: Se muestran el resumen de resultados de los modelos analizados. Datos no estandarizados. Elaboración propia.

Los modelos previamente revisados presentan valores de error por debajo del 5%; a partir de lo que se presume que puede existir sobreajuste en los modelos. Para identificar si los modelos tienen este comportamiento se hace una revisión utilizando la técnica de *validación cruzada de k iteraciones*. Se elige una *k* con valor diez, que es el más común utilizado en la literatura (Ghojogh & Crowley, 2019). Primero se revisa el modelo de red neuronal con tres capas ocultas: 50, 20 y 12 neuronas respectivamente; a partir del que se obtienen los resultados que se muestran en la Tabla 3-16. Como se aprecia, el error cuadrático medio que se obtiene utilizando los datos de validación para cada una de las diez iteraciones es considerablemente mayor que el error cuadrático medio que se obtiene al entrenar el modelo con los datos correspondientes. Este comportamiento es propio de un modelo con sobreajuste; es decir, el modelo no es capaz de hacer predicciones precisas con nuevos datos. Para disminuir o eliminar el sobreajuste se debe hacer un intercambio en sus características; en este caso, consiste en reducir la precisión del modelo a cambio de aumentar su

capacidad predictiva. Sin embargo, en el caso de los modelos planteados en el presente trabajo se consideran una solución local al problema.

Tabla 3-16

Error cuadrático medio (MSE) por iteración en la validación cruzada en modelo de regresión

	1	2	3	4	5	6	7	8	9	10
MSE entrenamiento	0.0232	0.0244	0.0267	0.0166	0.0244	0.0269	0.0319	0.0355	0.0252	0.0313
MSE validación	3.7394	2.9244	3.1398	2.5500	2.2100	2.7365	2.7162	2.9371	3.9912	3.2361

Nota: Error cuadrático medio (MSE) en el modelo de redes neuronales para la regresión en el precio de las acciones. Modelo con tres capas ocultas: 50, 20 y 12 neuronas respectivamente. Elaboración propia.

Se realiza la misma validación para el modelo de redes neuronales que clasifica el sentido de la variación en el precio de las acciones. La Tabla 3-17 muestra el porcentaje de precisión obtenida en cada una de las iteraciones de la *validación cruzada*. De igual forma que el modelo para la predicción, existe una diferencia significativa entre los valores de precisión obtenidos con los datos de entrenamiento, y con los datos de validación; por lo que se determina que el modelo que realiza la clasificación del sentido de la variación en el precio de las acciones es apropiado para su implementación directa en la herramienta como una solución local. Tomando en cuenta este hallazgo, es indispensable realizar un entrenamiento al modelo cada vez que se recolecten nuevos datos a la base de datos con el din de que las predicciones y clasificaciones correspondientes sean confiables. Así mismo, dado que los modelos arrojan valores estimados para un día posterior a la fecha en que se realiza la consulta, los modelos propuestos sirven como base para el planteamiento de una herramienta predictiva con datos en tiempo real; sin embargo, queda fuera del alcance de este proyecto y permanece como tema para futuros trabajos.

Tabla 3-17

Porcentaje de precisión por iteración en la validación cruzada en modelo de clasificación

	1	2	3	4	5	6	7	8	9	10
MSE entrenamiento	85.92	95.61	93.84	95.35	96.92	97.38	95.87	96.13	91.95	95.43
MSE validación	50.88	51.76	47.05	50.00	46.07	49.41	51.17	43.52	53.25	51.28

Nota: Error cuadrático medio (MSE) en el modelo de redes neuronales para la clasificación del sentido de la variación en el precio de las acciones. Modelo con tres capas ocultas: 120,120 y 100 neuronas respectivamente. Elaboración propia.

3.4.3 Modelo de clasificación mediante *agrupamiento por k medias*

En paralelo, el conjunto de datos de la tabla de hechos *Posición* contiene atributos categóricos derivados del análisis técnico: *Stochastic indicator* y *MACD_Signal_Cross*; el primero es una recomendación de compra o venta, y el segundo indica la tendencia en el precio. Los atributos sirven al inversionista como guía para tomar decisiones de inversión con base en el comportamiento histórico del instrumento financiero, por lo que entender su relación con los datos de las noticias financieras y de los indicadores económicos puede aportar ventajas a la hora de invertir.

En este proyecto se propone el modelo de aprendizaje automático no supervisado *agrupación por k-medias,* con el objetivo de agrupar los datos cuyas características sean similares e identificar las condiciones en el mercado bursátil bajo las que una inversión tiene más posibilidades de generar ganancias económicas. Debido a la que el conjunto de datos contiene diez atributos (*Score_News, Score_News_sq, VIX, EEM, TNX, CLF, ESF, GFC, USD/MXN, USD/EUR*), previo al modelado de k-medias se realiza una reducción de dimensión utilizando el método de *análisis de componentes principales* (PCA) para evitar el efecto conocido como *la maldición de la dimensionalidad* (Yiu, 2019).

El análisis de componentes principales (PCA) permite definir un espacio de dimensiones reducidas que preserva la información relevante de los datos originales; y está diseñado para

modelar datos que se caracterizan por una correlación no trivial entere las variables involucradas (Geladi & Linderholm, 2020); y tal es el caso del conjunto de datos de la tabla de hechos Posición, pues como se presentó en la Gráfica 3.10, la correlación entre el porcentaje de variación diaria (*% Variación*) y el resto de los atributos no se identifica de forma simple; lo mismo sucede con el sentimiento de las noticias financieras (*Score_News)*. A continuación, se presenta el algoritmo para el análisis de componentes principales desarrollado en *Python;* cabe destacar que el algoritmo se apoya de las herramientas que ofrece la librería de uso *scikit-learn*[14]:

```
# Adaptación realizada a partir de una publicación realizada por Galarnyk, Michael  el 04/12/2017
# en el sitio: https://towardsdatascience.com/pca-using-python-scikit-learn-e653f8989e60
pca = PCA(n_components=10)
principalComponents = pca.fit_transform(data)
principalDf = pd.DataFrame(data = principalComponents
        , columns = ['principal component 1'
                   , 'principal component 2'
                   , 'principal component 3'
                   , 'principal component 4'
                   , 'principal component 5'
                   , 'principal component 6'
                   , 'principal component 7'
                   , 'principal component 8'
                   , 'principal component 9'
                  , 'principal component 10'])

principalDf
finalDf = pd.concat([principalDf, true_label_stochastic], axis = 1)
finalDf = pd.concat([finalDf, true_label_names], axis = 1)
principalDf_array = principalDf.to_numpy()

components = pd.DataFrame(data    = pca.components_,
   columns = data.columns,
   index   = ['PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6', 'PC7', 'PC8', 'PC9', 'PC10']
)

eigenvalues = pd.DataFrame(data = pca.explained_variance_,
   columns = ['Eigenvalues'],
   index   = ['PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6', 'PC7', 'PC8', 'PC9', 'PC10']
)
proportion = pd.DataFrame(data = pca.explained_variance_ratio_,
   columns = ['Proportion'],
   index   = ['PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6', 'PC7', 'PC8', 'PC9', 'PC10']
)
eigen=eigenvalues.join(proportion)
eigen['Cumulative']=proportion.cumsum()
```

[14] Librería de uso libre que ofrece herramientas de aprendizaje automático en Python.

La Tabla 3-18 muestra los componentes principales con sus valores propios; como se puede ver, cinco de los componentes ayudan a explicar cerca del 91% de la variación en los datos; en este sentido, es factible trabajar con cinco variables (componentes), en lugar de los diez originales.

Tabla 3-18

Componentes principales y sus valores propios

Componente	Eigenvalue	Proporción	Acumulado
PC1	3.839182	0.351301	0.351301
PC2	2.133191	0.195196	0.546497
PC3	1.880394	0.172064	0.718561
PC4	1.178825	0.107867	0.826428
PC5	***0.951079***	***0.087028***	***0.913456***
PC6	0.504001	0.046118	0.959574
PC7	0.214275	0.019607	0.979181
PC8	0.134471	0.012305	0.991486
PC9	0.070039	0.006409	0.997895
PC10	0.023004	0.002105	1.000000

Nota: Se muestran nueve componentes principales con sus valores propios (*Eigenvalue*) representados de forma proporcional y acumulada. Elaboración propia.

A continuación, en la Tabla 3-19 se identifican las mayores asociaciones entre las variables y los componentes; estas asociaciones permiten asignar una interpretación a cada uno. Por ejemplo, el primer componente tiene una asociación negativa con *TNX* (Interés de los bonos de la tesorería a 10 años), *CLF* (Precio de petróleo a futuro), y positiva con *USD/MXN* (Tipo de cambio USD/MXN); a partir de lo que es posible interpretar que el primer componente mide principalmente la desconfianza en el mercado de valores de EE.UU.

Tabla 3-19

Vectores propios (Eigenvectors) de los cinco componentes principales seleccionados

Variable	PC1	PC2	PC3	PC4	PC5
Score_News	0.0221	0.0726	0.0218	-0.3741	***-0.9233****
VIX	0.3291	-0.0406	-0.0617	***0.5123****	-0.1764
EEM	-0.0297	***0.6204****	0.0314	-0.2513	0.1510
TNX	***-0.4103****	-0.0057	0.0558	-0.3462	0.1547

Variable	PC1	PC2	PC3	PC4	PC5
CLF	***-0.4097****	0.3317	0.0301	0.1445	-0.0344
USD/MXN	***0.4440****	-0.0019	-0.0162	-0.2776	0.1241
USD/EUR	0.2899	***-0.4275****	0.0031	***-0.4148****	0.1474
ESF	0.3528	***0.4007****	0.0014	-0.2891	0.1655
GFC	0.3773	0.3933	-0.0429	0.2367	-0.0626
Score_News_Sq	0.0784	-0.0157	***0.9943****	0.0704	-0.0045

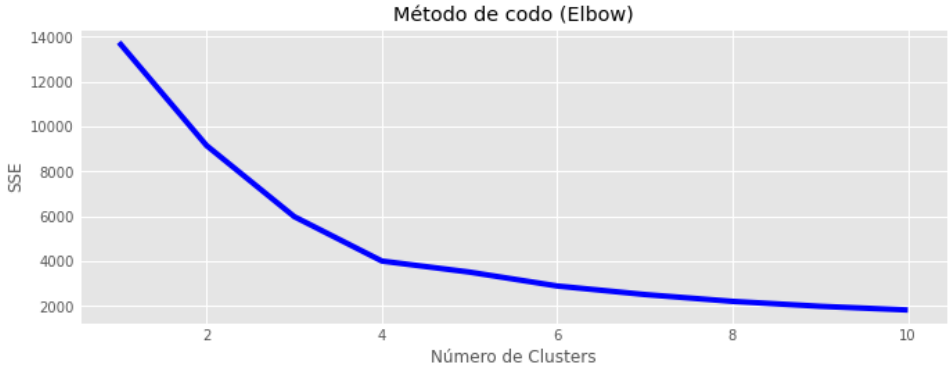
Nota: Se muestran cinco componentes principales con sus vectores propios (*Eigenvectors*). *Indica una asociación grande entre el componente y la variable. Elaboración propia.

A partir de los valores que se muestran en la Tabla 3-19, y con base en las representaciones indicadas en la Tabla 3-3 se asigna un significado conceptual a cada uno de los componentes como se enlista a continuación:

- Componente 1: Desconfianza en el mercado de valores EE.UU.
- Componente 2: Expectativas de crecimiento en el mercado global.
- Componente 3: Popularidad en los medios de comunicación.
- Componente 4: Volatilidad en el mercado de valores de EE.UU.
- Componente 5: Sentimiento de noticias financieras.

Una vez que se han identificado los componentes, se procede a seleccionar la cantidad óptima de clústeres en los que se agruparán los datos; para ello se utiliza el método de *codo*. Es un método heurístico que se basa en la idea de que el número *k* ideal de clústeres es aquél cuya adición de otro clúster no mejora significativamente el modelo; en este sentido, se asume que el primer clúster contribuye en mayor medida a la reducción de la suma del error cuadrático (SSE)[15], y del mismo modo los clúster subsecuentes; pero en un punto la contribución de cada clúster adicional se reduce drásticamente, de modo que en el gráfico de la suma del error cuadrático (SSE) en función del número de clústeres, se forma un ángulo que representa el *codo*, y que define el número óptimo de clústeres para el modelo (Bholowalia, 2014). En la Gráfica 3.25 se puede identificar que el punto en el que se forma el *codo* es cuatro, por lo que se toma como el número óptimo de clústeres para el modelo.

[15] La suma de las diferencias al cuadrado entre cada dato y el centroide de cada clúster (Huguenard, 2017)



Gráfica 3.25. Suma del error cuadrático (SSE) en función del número de clústeres. Elaboración propia

En la Tabla 3-20 se muestran las coordenadas de los cuatro centroides definidos utilizando el método de codo; la ubicación de los centroides en relación con los componentes principales es de ayuda para dar una interpretación a los datos contenidos en cada uno de ellos.

Tabla 3-20

Coordenadas de los centroides con respecto a los componentes principales

	PC1	PC2	PC3	PC4	PC5
C1	-3.230	1.131	-0.090	1.439	-0.478
C2	-0.002	0.712	0.149	-0.824	0.317
C3	4.004	1.225	-0.265	1.105	-0.322
C4	-0.167	-1.555	-0.029	-0.067	-0.037

Nota: Se muestran las coordenadas de los centroides con respecto a los componentes principales. Elaboración propia.

El código que se muestra a continuación ejecuta el método de *agrupamiento por k-medias*; con el que, a partir de los resultados del método de *codo* se calculan cuatro grupos (clústers), y se grafican respecto a los componentes principales *PC1*, *PC2*, *PC3* y *PC4:*

```
# Adaptación realizada a partir de una publicación realizada por Arvai, Kevin
# en el sitio:https://realpython.com/k-means-clustering-python/#how-to-perform-k-means-clustering-in-python

from sklearn import preprocessing
from sklearn.preprocessing import LabelEncoder, MinMaxScaler
from sklearn.cluster import KMeans

label_encoder = LabelEncoder()
true_labels = label_encoder.fit_transform(true_label_names)

kmeans = KMeans(
init="random",
```

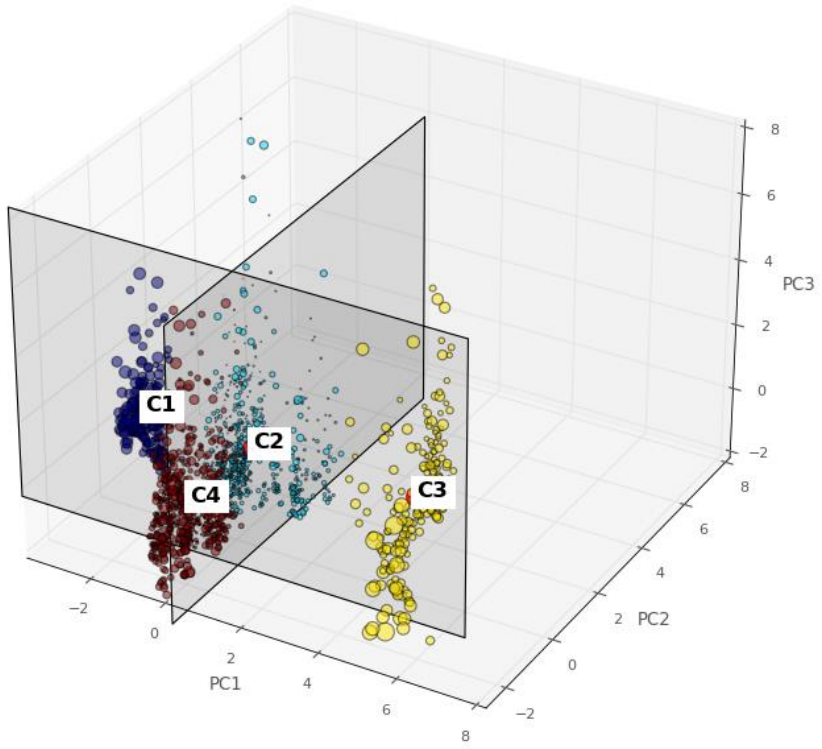
```
n_clusters=4,
n_init=1000,
max_iter=300
# ,random_state=42)

kmeans.fit(principalDf_array)
centroids=kmeans.cluster_centers_
centroides=pd.DataFrame(data=centroids,
   columns  = ['PC1','PC2','PC3','PC4','PC5','PC6','PC7','PC8','PC9','PC10']
   index   = ['C1','C2','C3','C4'])

kmeans_labels=pd.DataFrame(data=kmeans.predict(principalDf_array),
   columns   = ['Cluster'], #Integrar noticias
   index   = range(0,len(kmeans.predict(principalDf_array))))

true_kmeans_labels = label_encoder.fit_transform(kmeans_labels)
true_kmeans_labels_unique = label_encoder.classes_

df_true_kmeans_labels_unique = pd.DataFrame(data=true_kmeans_labels_unique,
   columns   = ['Cluster'], #Integrar noticias
   index   = range(0,len(true_kmeans_labels_unique)))

df_true_kmeans_labels_unique

#Código para graficar los datos y clústers en términos de los componentes principales
# Adaptación realizada a partir de una publicación realizada por Pythonprogramming
# en el sitio: https://pythonprogramming.net/3d-graphing-pandas-matplotlib/
from matplotlib import cm from matplotlib.colors import ListedColormap,LinearSegmentedColormap
from matplotlib.ticker import LinearLocator, FormatStrFormatter
from mpl_toolkits.mplot3d import Axes3D


x = np.array([[0, 0], [0, 0]])
y = np.array([[-2.6, -2.6], [8, 8]])
z = np.array([[-3, 6], [-3, 6]])

x1 = np.array([[-6, -6], [6, 6]])
y1 = np.array([[0, 0], [0, 0]])
z1 = np.array([[-3, 6], [-3, 6]])

Xlim = [-3.5,8]
Ylim = [-2.6,8]
Zlim = [-2,8]

scaler = MinMaxScaler()
PC4_array=principalDf['principal component 4'].to_numpy()
df_PC4 = pd.DataFrame(scaler.fit_transform(PC4_array.reshape(-1,1)), index=range(0, len(principalDf['principal
component 4'])))

threedee = plt.figure(figsize = (10,10)).gca(projection='3d')
threedee.scatter(principalDf['principal component 1'],
            principalDf['principal component 2'],
            principalDf['principal component 3'],
```

```
            c= kmeans_labels['Cluster'],
            s= 150*df_PC4.multiply(df_PC4),
            alpha=0.5,
            label=df_true_kmeans_labels_unique)

threedee.scatter(centroids_df['PC1'],
            centroids_df['PC2'],
            centroids_df['PC3'],
            s=200,
            alpha=0.8,
            color='r',
            zorder=25)

threedee.set_xlabel('PC1')
threedee.set_ylabel('PC2')
threedee.set_zlabel('PC3')
threedee.set_facecolor('w')
threedee.set_xlim(Xlim)
threedee.set_ylim(Ylim)
threedee.set_zlim(Zlim)

for i in range(len(centroides['PC1'])): #plot each point + it's index as text above
   threedee.text(centroides['PC1'][i],centroides['PC2'][i],centroides['PC3'][i],  '%s' % ('C'+str(i+1)), size=15,
zorder=200,
   color='k',backgroundcolor='w',fontweight='1000')
plt.show()
```

La Gráfica 3.26 permite visualizar la distribución de los datos de los primeros cuatro componentes principales, que en conjunto explican el 82.64% de la variación de los datos; los primeros tres componentes se muestran en los ejes *PC1, PC2* y *PC3*; mientras que el cuarto componente se visualiza en el tamaño de los datos. Como se observa, los datos que se concentran en valores de *PC1* cercanos a cero, tienen un valor de *PC4* bajo; independientemente de los valores de *PC2* y *PC3*; a partir de lo que es posible interpretar que generalmente, cuando la desconfianza en el mercado de valores (*PC1*) es baja, la volatilidad se mantiene estable (*PC4*) a pesar de las expectativas de crecimiento económico (*PC2*) y la popularidad del instrumento en los medios de comunicación (*PC3*). En este sentido, se aprecia que los grupos *C1* y *C3* cuyos valores indican grados de desconfianza (*PC1*) en los extremos positivos y negativos, presentan valores de volatilidad en el mercado (*PC4*) más altos.

Los clústeres *C2* y *C4*, se diferencian entre sí principalmente por las expectativas de crecimiento económico (*PC2)* y el sentimiento de las noticias financieras (*PC5*); el clúster *C2* al contrario que el *C4* agrupa datos cuyas noticias y expectativas de crecimiento son positivas; a partir

de ello, es posible afirmar que el clúster *C2* corresponde al momento en el mercado en el que existe confianza (sin exceso), expectativas de crecimiento y noticias positivas; y en consecuencia baja volatilidad; bajo esas condiciones no se esperan variaciones drásticas en el precio de las acciones.



Gráfica 3.26. Distribución de los datos de los componentes principales *PC1*, *PC2*, *PC3, PC4* y los centroides de cada uno de los cuatro grupos (*C1*, *C2*, *C3* y *C4*). Se muestran datos correspondientes a la acción *AAPL*. Elaboración propia

Como se mencionó anteriormente, la identificación de los grupos además de permitir encontrar relaciones entre los datos; es útil para identificar las condiciones bajo las que es más probable generar rendimientos positivos de una inversión; la Tabla 3-21 muestra un resumen de los clústeres calculados mediante el método de agrupamiento por *k-medias* en relación con los atributos *Stochastic indicator* y *MACD Signal Cross* calculados previamente. Se observa que los clústeres *C2* y *C4* son los que contienen una mayor cantidad de datos cuyos valores para el atributo *Stochastic indicator* son sobrecompra (*Overbought*) y venta (*Sell*); en ambos casos el indicador *MACD_Signal* es *Up_Over 0*. Así mismo, son los únicos clústeres en los que predomina el valor de compra (*Buy*) y sobreventa (*Oversold*); y también el valor del atributo *MACD_Signal* es

mayoritariamente "*Down_Below_0*". A partir de la observación previa, es posible afirmar que el mejor momento para comprar y vender en el mercado de valores es cuando las condiciones se asemejan a las de los clústeres *C2* y *C4*. En conjunto los clústeres *C2* y *C4* contienen el 74% de los datos.

Respecto a los clústeres *C1* y *C3*, aunque también agrupan datos cuyos atributos *Stochastic indicator* y *MACD_Signal* son *Sell Overbougth* y *Up_Over_0* respectivamente; su proporción es más baja respecto a lo clústeres *C2* y *C4*, y no se identifica otra tendencia importante en relación con los atributos mencionados; además aunado a que en estos clústeres los datos presentan mayores valores de volatilidad, se considera que cuando las condiciones del mercado se asemejan a las de los clústeres *C1* y *C3*, no es el mejor momento para realizar transacciones en el mercado. Los clústeres *C1* y *C3* en conjunto agrupan el 26% de los datos.

Tabla 3-21

Relación entre indicadores por clúster

Stochastic_indicator MACD_Signal	**C1**	**C2**	**C3**	**C4**	**Total**
Buy	***0.53%***	***2.22%***	***0.70%***	***2.92%***	***6.37%***
C_Down	0.00%	0.06%	0.00%	0.00%	0.06%
C_Up	0.00%	0.06%	0.00%	0.06%	0.12%
Down_Below_0	0.35%	**1.75%**	0.58%	**2.16%**	4.85%
Down_Over_0	0.18%	0.29%	0.12%	0.64%	1.23%
Up_Below_0	0.00%	0.06%	0.00%	0.06%	0.12%
Hold	*6.96%*	*15.91%*	*4.56%*	*16.37%*	*43.80%*
C_Down	0.35%	0.82%	0.35%	0.76%	2.28%
C_Up	0.23%	0.70%	0.23%	0.76%	1.93%
Down_Below_0	1.17%	2.16%	0.94%	3.74%	8.01%
Down_Over_0	2.92%	7.02%	1.40%	5.85%	17.19%
Up_Below_0	1.40%	1.81%	0.47%	3.10%	6.78%
Up_Over_0	0.58%	3.22%	1.11%	1.75%	6.67%
Zero Cross_Downwards	0.12%	0.12%	0.06%	0.23%	0.53%
Zero Cross_Upwards	0.18%	0.06%	0.00%	0.18%	0.41%
Overbougth	*2.92%*	*9.42%*	*3.80%*	*7.13%*	*23.27%*
C_Down	0.00%	0.06%	0.00%	0.06%	0.12%
C_Up	0.29%	0.53%	0.18%	0.18%	1.17%
Down_Over_0	0.23%	0.82%	0.35%	0.76%	2.16%
Up_Below_0	0.12%	0.82%	0.41%	1.23%	2.57%
Up_Over_0	2.22%	**6.84%**	2.81%	**4.56%**	16.43%
Zero Cross_Upwards	0.06%	0.35%	0.06%	0.35%	0.82%
Oversold	0.58%	***2.75%***	0.94%	***4.15%***	***8.42%***
C_Down	0.00%	0.00%	0.00%	0.12%	0.12%
Down_Below_0	0.29%	**1.40%**	0.58%	**2.34%**	4.62%
Down_Over_0	0.23%	0.82%	0.23%	1.29%	2.57%
Up_Below_0	0.00%	0.29%	0.00%	0.23%	0.53%
Zero Cross_Downwards	0.06%	0.23%	0.12%	0.18%	0.58%
Sell	***2.51%***	***7.02%***	***2.46%***	***6.14%***	***18.13%***
C_Down	0.18%	0.35%	0.00%	0.06%	0.58%
Down_Over_0	0.23%	0.82%	0.12%	0.82%	1.99%
Up_Below_0	0.06%	0.06%	0.06%	0.76%	0.94%
Up_Over_0	2.05%	**5.73%**	2.16%	**4.50%**	14.44%
Zero Cross_Upwards	0.00%	0.06%	0.12%	0.00%	0.18%
Total	**13.51%**	**37.31%**	**12.46%**	**36.73%**	**100.00%**

Nota: Se muestran la relación entre los indicadores *MACD_crossover* y *Stochastic_indicator* para cada uno de los clústeres con respecto a los componentes principales. Elaboración propia.

En este capítulo se llevó a cabo el desarrollo del proyecto, que de acuerdo con la metodología CRISP-DM propuesta, comprende las fases: *comprensión de los datos, preparación de los datos,* y *modelado y evaluación.* En la fase de comprensión, se realizó la recopilación de los datos transaccionales históricos de las cincuenta acciones más representativas del índice bursátil *S&P500,* y los datos de los índices bursátiles; ambos del sitio *Yahoo Finance;* también, se recopilaron datos de las noticias financieras en torno a las cincuenta acciones; en este caso del sitio *Google*. La recopilación de los datos se llevó a cabo mediante APIs disponibles en el repositorio público para Python (*Pyhon Package Index*); y con ayuda de métodos de extracción de datos de sitios web, conocidos popularmente por su nombre en inglés *Web Scrapping*.

Una vez recopilados los datos, se exploraron para evaluar su calidad, identificar problemas en su contenido y estructura; enseguida se procedió con la fase de *prepararon los datos*, en la que se transformaron y se estandarizaron con el propósito de generar tres conjuntos de datos que se pudieran relacionar entre sí: *datos transaccionales, noticias financieras* y *índices económicos.* Por último, se calcularon atributos derivados y se eliminaron algunos registros que podían causar errores en la fase subsecuente y se cargaron en una base de datos multidimensional en *MySQL* que relaciona los tres conjuntos de datos en una tabla de hechos. Estas dos primeras fases representaron el 80% del trabajo correspondiente a este capítulo.

3.5 Implementación

La fase de implementación consiste en generar una herramienta a través de la que el usuario sea capaz de obtener un resumen de los resultados del análisis y modelado de los datos, de forma que pueda tomar decisiones basadas en datos. En este proyecto se propone la integración de los códigos que se desarrollaron en la fase previa, como subfunciones para que, de forma automática, y a partir de parámetros dados por el usuario se ejecute la recolección, transformación, carga y modelado de datos; así como la evaluación de los resultados; y se genere un reporte en el que se indique la predicción del porcentaje de variación de cada una de las cincuenta acciones, y las condiciones actuales en el mercado de valores.

La Figura 3.12 muestra el diagrama de flujo del proceso para generar el reporte. El usuario proporciona como datos de entrada la fecha (*Fecha_U)* y el código bursátil (*Ticker)* para los cuales

desea obtener la información; enseguida se hace una consulta a la tabla de hechos *Posición* de la base de datos, tomando como criterio de búsqueda el valor *Ticker*; en el caso de que el valor de *Fecha_U* sea menor o igual que la fecha del último registro de la consulta, se genera un reporte a partir de los resultados generados en la fase de desarrollo; en caso contrario, se ejecutan las subrutinas que hacen la recolección de los datos de las noticias financieras, datos transaccionales de las acciones, y los datos de los índices económicos. Así mismo, una vez recolectados los datos, se limpian, se transforman y se calculan los atributos derivados, para posteriormente cargarlos en la base de datos. Para finalizar, dado que se han cargado nuevos datos, se ejecutan las subrutinas para el modelado de la red neuronal artificial y el *agrupamiento por k medias.* A partir de los resultados obtenidos se genera un reporte que se muestra al usuario con el propósito de que pueda tomar mejores decisiones de inversión

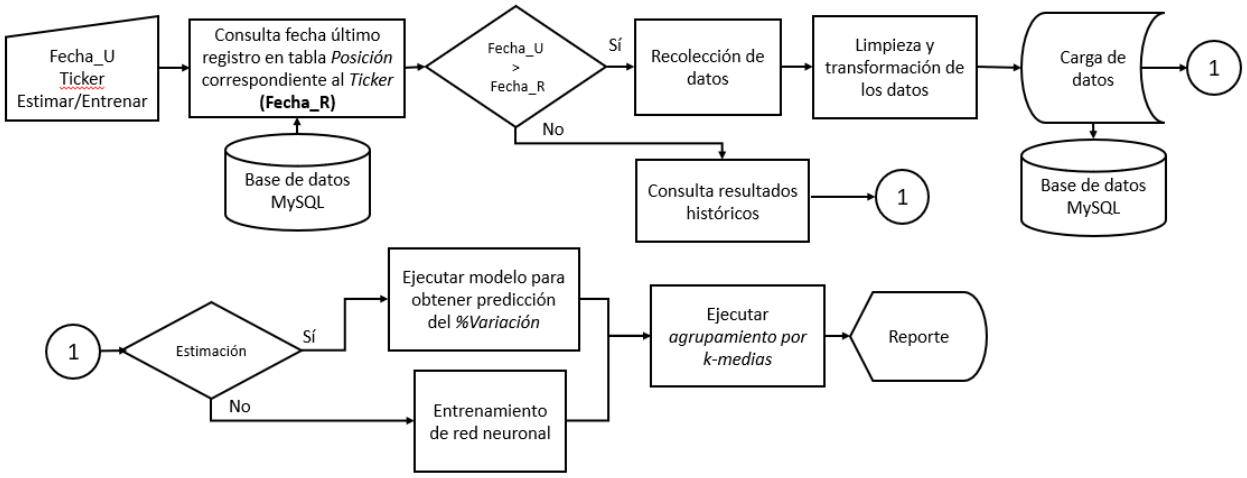


Figura 3.12. Diagrama de flujo del proceso de implementación.

Para poder realizar la implementación, es necesario asegurar la integración del código en *Python*, *Matlab* y *MySQL* editor; la conexión entre *Python* y *MySQL*, se estableció previamente en la fase de desarrollo del proyecto; pero hasta el momento no existe una conexión entre *Python* y *Matlab*; para ello, se utiliza la API *Matlab Engine*[16] que permite ejecutar scripts[17] en *Matlab* a través de *Python.* Se establece comunicación entre los tres softwares tal como se esquematiza en

[16] API disponible en Matlab que permite ejecutar scripts de (The Mathworks, Inc., 2021)
[17] Código que ejecuta diversas tareas que en conjunto se pueden considerar una aplicación (Ousterhout & K., 1998)

la Figura 3.13 a modo de que los datos puedan fluir entre ellos y sea posible generar un reporte de forma automática utilizando *Jupyter Notebooks* como única interfaz para el usuario.

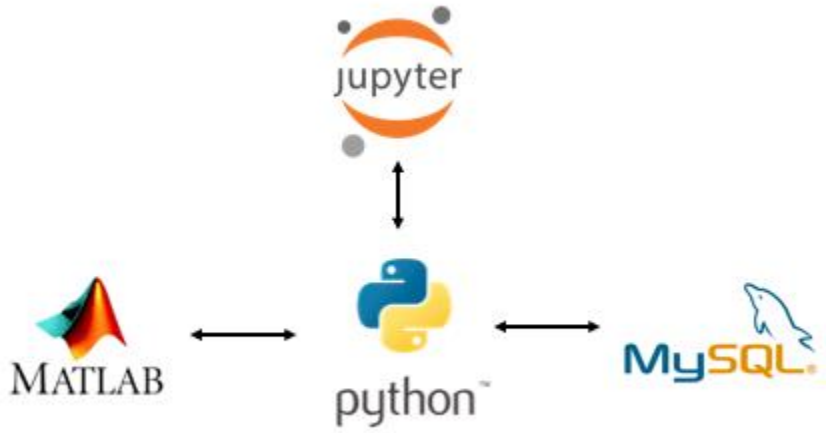


Figura 3.13. Conexión entre Python, Matlab y MySQL

También se hace uso de las librerías *voila*[18] y *ipwidgets*[19] que proporcionan herramientas para crear una aplicación interactiva. El propósito de la aplicación es que el usuario sea capaz de ejecutar la herramienta y realizar consultas de forma simple, sin la necesidad de tener que interactuar directamente con el código. En este caso, la implementación se lleva a cabo en el sistema operativo Windows; esto ofrece la posibilidad de ejecutar la aplicación desde un archivo con formato *.bat*[20] utilizando el código que se muestra a continuación:

```
@ECHO OFF
set ruta=C:/Users/eidri/
ECHO Ejecutando aplicacion....

call %ruta%anaconda3/Scripts/activate.bat
call conda activate py370
call voila %ruta%Test.ipynb

PAUSE
```

EL archivo *.bat* funciona como un acceso directo a la aplicación, al ejecutarlo se abre el explorador de internet predeterminado y se muestra la aplicación en la que el usuario ingresa la fecha y el código bursátil, para los cuales desea hacer la consulta. La Figura 3.14 exhibe la pantalla

[18] Permite convertir un cuaderno de Jupyter en una aplicación autónoma (Voila-Gallery, 2021)
[19] Contiene herramientas interactivas para Jupyter Notebooks (Project Jupyter Revision, 2021)
[20] Archivo que ejecuta rutinas de comandos en la consola de Windows de forma automática. (Microsoft, 2021)

que se visualiza al ejecutar la aplicación. Se presentan dos campos interactivos de entrada de datos: *Start Date* y *Ticker*. El campo *Start Date* abre un calendario para que sea posible seleccionar la fecha de consulta; en caso de que la fecha sea posterior a la fecha en que se realiza la consulta, se muestra un mensaje de error que indica que la fecha es incorrecta. El campo *Ticker* despliega una lista con los códigos bursátiles que están disponibles para realizar la consulta; en caso de que el código bursátil introducido no coincida con alguno de los que se muestran como disponibles, se muestra un mensaje de error indicando que el código bursátil es incorrecto. La integración de los campos de entrada de datos interactivos además de que facilitan al usuario el uso de la aplicación; limitan la entrada de datos para evitar que al ejecutar las subrutinas se generen errores que impidan finalizar el proceso para la generación del reporte.

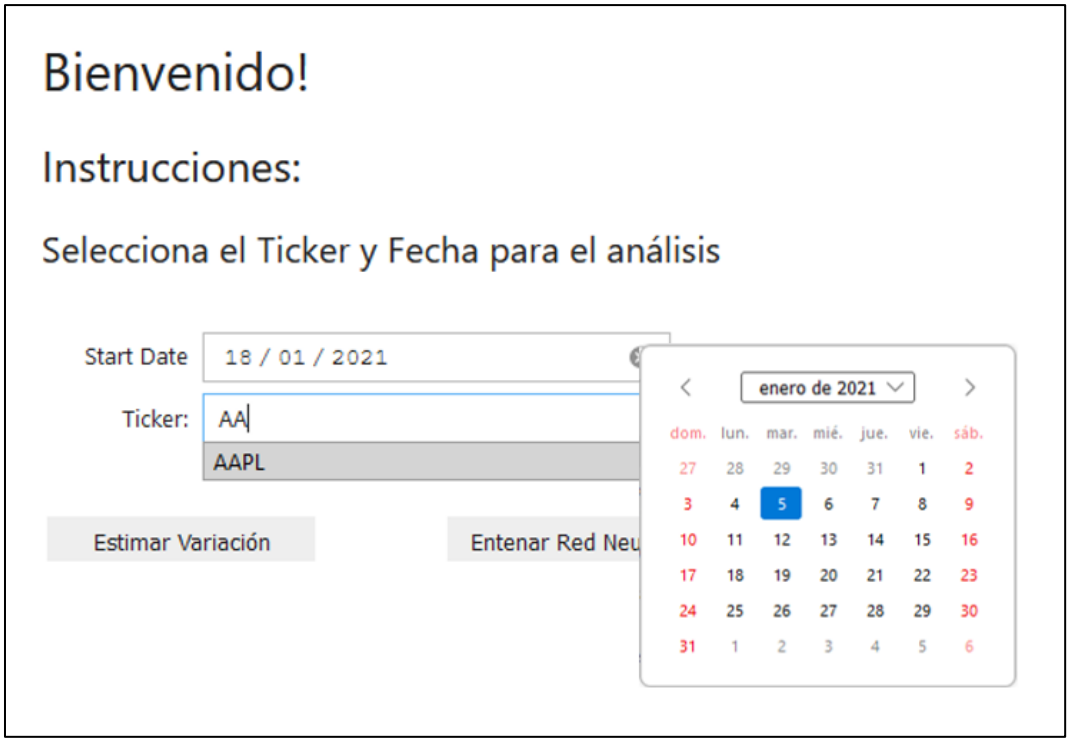


Figura 3.14. Visualización de la aplicación y los campos de entrada de datos.

Una vez que el usuario ingresa los datos de entrada, procede a seleccionar entre las opciones *Entrenar Red Neuronal* y *Estimar Variación*. Enseguida se ejecutan las subrutinas de acuerdo con el proceso esquematizado en la Figura 3.12. La primera tarea es realizar una consulta a la base de datos de MySQL para obtener la fecha del último registro y compararla con la fecha introducida por el usuario; si la fecha dada por el usuario es anterior, se obtienen los datos de forma directa de

la base de datos; en caso contrario, se procede a ejecutar la subrutina que ejecuta la recolección de los datos. En la aplicación se muestran al usuario los pasos que se realizan durante el proceso con el fin de que pueda visualizar como se lleva a cabo el proceso, y así ofrecer la posibilidad de verificar que la aplicación se está ejecutando sin problemas. En la Figura 3.15 se presenta un ejemplo de lo que visualiza el usuario mientras se realiza la recolección de los datos. Cabe destacar que en este proceso se hace la recolección de todos los datos; es decir, las noticias financieras de *Google News*, los datos transaccionales, y los datos de los índices económicos de *Yahoo Finance*; por lo que el tiempo de ejecución dependerá de la cantidad de datos que se recolecten; y esto a su vez depende de qué tan grande sea la diferencia entre la fecha de los datos disponibles en la base de datos, y la fecha ingresada por el usuario.

Figura 3.15. Mensajes que se muestran al usuario durante el proceso de recolección de datos.

Durante la recolección de los datos se crean archivos en formato *.csv*[21] en los que se almacenan los datos recolectados de forma temporal y que sirven para realizar los siguientes pasos del diagrama de proceso de la Figura 3.12; es decir, la limpieza, transformación y carga de datos en la base de datos multidimensional. En la aplicación estas tareas suceden sin ofrecer muchos detalles al usuario y solo se muestran mensajes que indican el inicio y fin del proceso, con la finalidad de que el usuario conozca en que parte del proceso se encuentra. La Figura 3.16 muestra un ejemplo de los mensajes que se imprimen en la pantalla de la aplicación durante la limpieza, transformación y carga de datos.

```
Inicia carga a base de datos.....
Carga de datos finaliza con éxito
Proceso ETL finalizado con éxito
```

Figura 3.16. Mensajes que se muestran al usuario durante el proceso de transformación, limpieza y carga de datos en la base de datos multidimensional

Ya que los datos están preparados y disponibles en la base de datos en *MySQL,* se establece comunicación con *MatLab* para transferir los datos al algoritmo del modelo de red neuronal. El modelo de red neuronal se ejecuta desde *Python* en *MatLab* por lo que los resultados obtenidos en *Matlab* se trasladan a *Python* para que se puedan procesar en la aplicación. El despliegue de los resultados depende de la opción de análisis seleccionada por el usuario. En el caso de la opción *Estimar Variación,* se muestra simplemente el valor estimado del porcentaje de variación; si es que la fecha ingresada inicialmente por el usuario es anterior a la fecha en que realiza la consulta, se muestra también el porcentaje de variación real como se aprecia en la Figura 3.17. Por el contrario, si la opción seleccionada por el usuario es *Entrenar Red Neuronal,* como resultado de este proceso se obtiene el error cuadrático medio del modelo de red neuronal; así como el porcentaje de variación diaria estimada por el modelo para la acción (código bursátil) y la fecha introducidos por el usuario. También se muestra un gráfico del porcentaje de variación diaria con respecto a la variable *Score_News_Sq* para los datos históricos, y los datos estimados por la red neuronal; y un gráfico del error cuadrático medio (MSE) para cada una de las épocas durante el proceso de entrenamiento de la red neuronal.

[21] Comma separated values: Archivos de texto que usan comas para separar valores (Johnson, 2021)

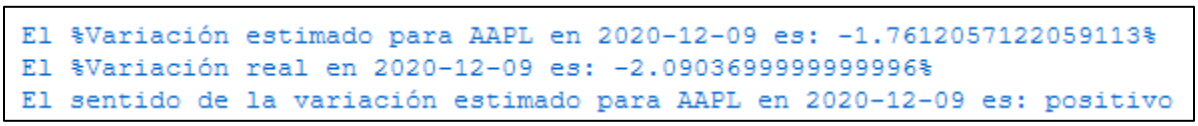


Figura 3.17. Ejemplo del resultado que arroja la aplicación al seleccionar la opción *Estimar Variación*

En paralelo, con base en los resultados del análisis de *agrupamiento por k medias*, se muestra al usuario la situación del mercado bursátil en la fecha de consulta. En la Figura 3.18 se aprecia un ejemplo de la visualización; en la que se hace una breve descripción de la situación en que se encuentra el mercado bursátil; así como los indicadores *Stochastic* y *MACD_Crossover*. También se presentan cuatro indicadores gráficos que ayudan al usuario a identificar de forma visual el nivel de desconfianza en el mercado, de las expectativas de crecimiento, de la popularidad de la acción en los medios de comunicación, y de la volatilidad. Estos indicadores son barras que representan el promedio de los datos de los diez días anteriores a la fecha de la consulta; el color rojo en las barras indica que en promedio los datos se encuentran por debajo de la media. En este caso, el número de días se eligió de forma arbitraria; sin embargo, el usuario puede modificar este valor de acuerdo con sus necesidades mediante un campo de entrada que le permite actualizar los gráficos de forma dinámica. Por último, se muestran dos gráficos con la distribución de los datos en términos de sus componentes principales; en el primero se visualiza la totalidad de los datos que corresponden a la acción, y se resalta la ubicación de los datos de los diez días anteriores a la fecha de la consulta. En el segundo gráfico el usuario puede ver el detalle de los datos resaltados en el primer gráfico. El propósito de estas vistas es que el usuario pueda interpretar la situación actual del mercado con mayor facilidad, y con eso sustentar sus decisiones de inversión.

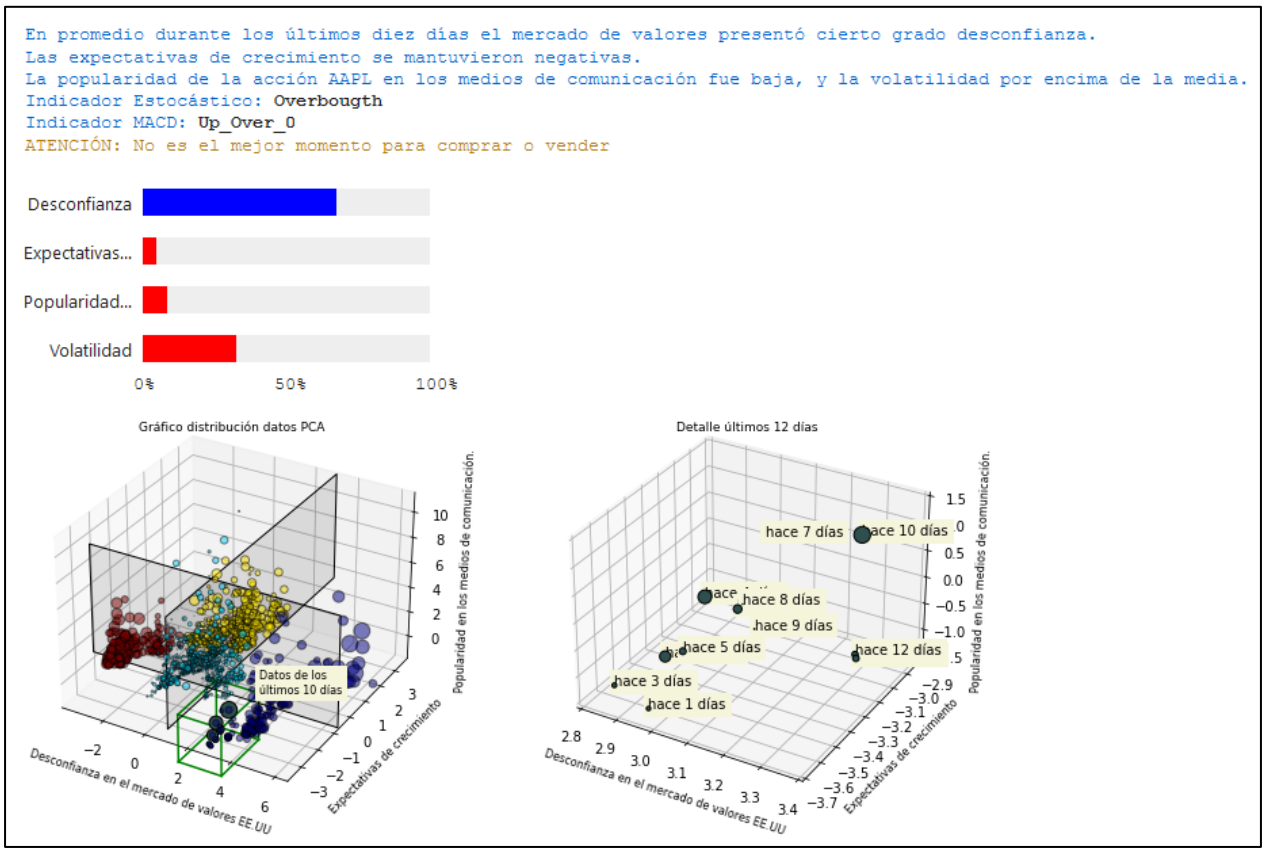


Figura 3.18. Vista del resultado de análisis de agrupamiento por k medias. Elaboración propia

4 Análisis y discusión

Apegado al objetivo general, en este proyecto se desarrolló una herramienta para apoyar la toma de decisiones de inversión basada en datos históricos transaccionales de las cincuenta acciones más representativas del índice S&P 500, noticias financieras e indicadores económicos. La herramienta permite obtener una estimación del porcentaje de variación en el precio de una acción en una fecha determinada; también determina las condiciones en el mercado de valores y a partir de ellas valora si es un buen momento para realizar transacciones.

La metodología CRISP-DM aplicada en el proyecto contribuyó al entendimiento del problema y consecuentemente de los datos; las dos primeras fases: comprensión del negocio y comprensión de los datos, son relevantes para llevar a cabo las últimas tres fases: preparación de los datos, modelado y evaluación, e implementación. Gran parte de la compresión del negocio se logró durante la revisión de la literatura y de antecedentes; al mismo tiempo se identificó que, aunque existen diversos estudios que aplican técnicas de aprendizaje automático para hacer predicciones en el mercado de valores; no se encontró evidencia de alguno cuya metodología, la integración de procesos, y los resultados sean similares a lo propuesto en este proyecto.

Las últimas tres fases de la metodología corresponden al desarrollo del proyecto, en la que se ejecutan procesos de recolección, limpieza, transformación y carga de datos. La recolección de los datos de las noticias financieras de *Google News* se logró utilizando un servicio VPN[22] que ayuda a evitar el bloqueo de las consultas mediante la rotación de la de la dirección IP (Internet Protocol). Este paso durante el proceso de recolección hace que incremente el tiempo de consulta ya que previo a realizar la recolección desde *Google News*, se inicializa el servicio VPN y se verifica que la dirección IP asignada no se encuentre bloqueada; en caso de que sí, se rota hasta que la consulta se pueda realizar. La recolección de datos de *Yahoo Finance*, aunque se realiza utilizando el mismo servicio VPN, es posible realizarla de forma independiente ya que *Yahoo* no limita el número de consultas.

[22] Virtual Private Network: es un método para establecer una conexión segura entre una computadora y una red externa. (University of Edinburgh, 2020)

La preparación de los datos implica la limpieza y transformación para asegurar que no hay datos faltantes, incongruentes o incompletos. A pesar de que la recopilación de noticias financieras de *Google News* se hizo utilizando como filtro de entrada la fecha y el idioma; se obtuvieron noticias de fechas e idiomas diferentes. Se procedió a identificar las noticias cuyo idioma fuera diferente al inglés y cuya fecha no se encontraba dentro del periodo entre 2014 y 2020; posteriormente se realizó la transformación mediante herramientas de procesamiento de lenguaje natural que categorizan las noticias como positivas, neutras o negativas. Respecto a los datos transaccionales de las acciones, e indicadores económicos recolectados de *Yahoo Finance* cuyo origen es el mercado bursátil, no presentan anomalías. La transformación de los datos de *Yahoo Finance* considera, entre otras cosas, el cálculo de atributos derivados basados en el análisis técnico propio del mercado de valores.

La carga de los datos se realiza en una base de datos multidimensional en MySQL que consiste en tres dimensiones: *datos transaccionales, sentimientos, índices económicos*, y una tabla de hechos: *posición*; el modelo multidimensional propuesto relaciona los conjuntos de datos utilizando la fecha y código bursátil de las acciones (*Ticker*) como llaves foráneas. Esta definición resulta importante ya que con ella se establece la forma en que se relacionan los datos; así como el tipo de consultas que se pueden realizar a la base de datos. La arquitectura definida es determinante para la implementación de la herramienta durante la última fase de la metodología CRISP-DM; en este punto se visualiza que por lo menos, la herramienta debe tener dos campos de entrada que corresponden a las llaves que relacionan los tres conjuntos de datos en el modelo multidimensional.

Inicialmente, la base de datos contiene los registros entre 2014 y 2020; cada vez que el usuario realiza una consulta en la herramienta con una fecha posterior al 31 de diciembre de 2020; se realiza la recolección, limpieza, transformación y carga en la base de datos; de esa forma se mantiene actualizada según las necesidades del usuario. La ventaja de esta forma de proceder es la optimización de recursos, pues solo se ejecuta el proceso para las acciones y fechas que el usuario indica; la desventaja es que, dependiendo de la actividad del usuario, pueden existir acciones que permanezcan desactualizadas por largos periodos de tiempo; y cuando se realice una consulta, la actualización puede consumir más tiempo en comparación con las acciones cuyos datos se encuentran actualizados en la base de datos.

El modelado se realiza utilizando los datos numéricos de la tabla de hechos *posición* del modelo multidimensional; estos datos son previamente estandarizados, ya que no se encuentran en las mismas unidades; por ejemplo, los sentimientos de las noticias financieras cuyos valores se encuentran expresados como porcentaje de positividad (-1: noticia negativa, y 1 noticia positiva) no se pueden procesar directamente con los datos del tipo de cambio USD/MXN cuyo valor entre 2014 y 2020 se encontró en el rango de 12.8372 y 25.3362. En este ejemplo, debido a que el valor de los datos del tipo de cambio es entre 5 y 10 veces mayor, tendrían mayor relevancia en el modelo que los datos del sentimiento de las noticias financieras. Para evitar este efecto, los datos se estandarizan para que sean equivalentes y el modelo arroje resultados adecuados.

De la exploración de los modelos de redes neuronales revisados para realizar la regresión del porcentaje de variación diaria en el precio de las acciones, y clasificación del sentido de la variación se distingue que existe un efecto positivo en la precisión al incrementar el número de neuronas en las primeras capas, y manteniendo el número de neuronas en las capas subsecuentes, por debajo del valor de la primera capa oculta. El modelo con 50,20 y 12 neuronas en sus tres capas ocultas respectivamente se identificó como el que tiene el mejor comportamiento, pues arroja el valor de error cuadrático medio (MSE) más bajo 0.0310. Se identificó que el tiempo de ejecución del algoritmo de entrenamiento se ve afectado por la complejidad de la red neuronal; es decir, a mayor cantidad de capas y neuronas, mayor es el tiempo de procesamiento; sin embargo, dado que el objetivo del proyecto no está relacionado con la optimización el proceso; no se tomó en cuenta este factor durante el desarrollo. Por último, cabe resaltar, que la aplicación del modelo de red neuronal para regresión y clasificación de forma combinada otorga al usuario inversionista más información que le puede servir para tomar mejores decisiones de inversión. El alto nivel de precisión del modelo (96.4%) para la clasificación del sentido de la variación en el precio de las acciones, respalda los resultados del modelo para la regresión del porcentaje de variación cuando ambos coinciden. Aun cuando los modelos de redes neuronales para predicción y clasificación parecen ofrecer resultados con un elevado nivel de precisión, al realizar la validación utilizando el método de *validación cruzada de k iteraciones* se encontró que los modelos presentan un sobreajuste; es decir, no son capaces de obtener los mismos niveles de precisión al procesar nuevos datos. En este contexto, los modelos propuestos solo representan una solución local al problema,

por lo que, para obtener una predicción y clasificación confiables, es recomendable reentrenar los modelos cada vez que se ingresen nuevos datos a la base de datos.

Para la identificación de las condiciones en el mercado de valores a partir del *agrupamiento por k medias*, se hace una reducción dimensional utilizando la metodología PCA (Principal Component Analysis); de esta forma se obtiene un conjunto de datos con cinco variables (componentes) que representan al conjunto de datos inicial correspondiente a la tabla de hechos *posición* de la base de datos multidimensional. La interpretación de los cinco componentes se basa en la magnitud de sus vectores propios (*eigenvector*s); es decir, el grado de asociación entre cada variable y el componente. En este sentido, se identifican las variables cuyos valores son más representativos para cada componente y se da una interpretación a cada uno, con base en el comportamiento propio de la interacción entre las variables. Se identificó que el número óptimo de clústeres es cuatro; a partir de los que derivan dos condiciones en el mercado de valores: *buen momento para la compra/venta de acciones* y *momento menos apropiado para la compra/venta de acciones.* La identificación de estos escenarios en el mercado de valores permite al usuario medir el riesgo y tomar decisiones de forma cautelosa.

La fase de implementación del proyecto representa la integración de los algoritmos y procesos desarrollados durante las fases de preparación de los datos, modelado y evaluación. Con fines didácticos, estas fases se desarrollaron utilizando diversas herramientas de manera independiente. La recopilación, limpieza, transformación y carga de datos se realizó en *Python* y *MySQL* a través de *Jupyter Notebooks;* mientras que el algoritmo para el modelo de redes neuronales se ejecutó en *Matlab.* Por esta razón fue necesario establecer una conexión entre las diversas herramientas para desarrollar una herramienta que permite al usuario generar un reporte en el que, a partir de una fecha y un código bursátil, se muestra el porcentaje estimado de variación en el precio; el sentido de la variación, y las condiciones en el mercado. La herramienta funciona de forma local y fue configurada de forma personalizada; sin embargo, es posible realizar una implementación utilizando un servicio PaaS (Platform as a Service) a modo de que se puedan realizar consultas en línea y no únicamente de forma local.

La herramienta desarrollada ofrece al inversionista una forma de respaldar sus decisiones, y de ese modo reducir el factor psicológico; en otras palabras, el inversionista tiene la posibilidad de

que sus decisiones no estén basadas en emociones y creencias, sino en lo que realmente está sucediendo en el mercado. Haciendo referencia a las finanzas conductuales, y a la hipótesis del mercado eficiente, se asume que las personas actúan con racionalidad procesal[23], mientras que el mercado de valores actúa con racionalidad sustantiva[24]. En este contexto es posible afirmar que el inversionista actúa basado en su experiencia y emociones, mientras que el mercado se comporta de forma eficiente respondiendo a todos los eventos que acontecen en él (Ritter, 2003). La herramienta propuesta en este proyecto es una forma de paternalismo libertario[25]; y en lugar de limitar al usuario, amplia sus posibilidades al momento de tomar decisiones; al final es siempre el inversionista quien ejecuta la operación en el mercado de valores.

[23] Propuesto por Herbert Simon en 1955, es un tipo de razonamiento que se basa en la heurística para la toma de decisiones (Dhami, al-Nowaihi, & Sunstein, 2019).
[24] Razonamiento que se basa en la maximización de los objetivos y de resultados (Dhami, al-Nowaihi, & Sunstein, 2019).
[25] Idea que defiende la legitimidad de influir en las conductas de los individuos siempre y cuando esto no anule su libertad de elección. (Zapata Claveria, 2015)

5 Conclusiones

El presente trabajo integra las fases del ciclo de vida de un proyecto de ciencia de datos, que de forma generalizada son reconocidos en el área de estudio[26]: Comprensión del negocio, Recolección de datos, Preparación de datos, Modelado, Evaluación, e Implementación. El ciclo completo se realizó utilizando *Python (Jupyter Notebooks)* como base del proyecto ya que gracias a su popularidad y a que es una herramienta de código abierto, tiene una amplia comunidad de usuarios que contribuye al desarrollo y publicación de nuevas librerías y aplicaciones (Voskoglou, 2017). (Pedregosa, y otros, 2011; Eastwood, 2020). De forma paralela, se hizo uso del manejador de bases de datos *MySQL* para realizar el almacenamiento de datos en una base de datos relacional a la cual se accede desde *Python* con ayuda de librerías disponibles en el repositorio de librerías de Python (PyPI). Por otra parte, en *MatLab* se hizo parte del modelado y la evaluación. Se eligió *MatL*ab debido a que su lenguaje de programación expresa matemática matricial de forma directa y simplifica el modelado de redes neuronales (The Mathworks, Inc., 2021). En este sentido, la integración de tres herramientas informáticas, así como la aplicación de diversas librerías agrega valor, y es un antecedente dentro del área de estudio.

Al igual que la mayoría de la literatura que se revisó para el desarrollo de este trabajo, se concluye que la integración de noticias financieras es un factor que ayuda a mejorar significativamente la precisión de los modelos predictivos. Este hallazgo, además contribuye a respaldar la hipótesis del mercado eficiente (EMH), que plantea que el mercado responde de forma racional a los eventos que acontecen a su alrededor; es decir, que la integración de una mayor cantidad y diversidad de datos (noticias financieras) es relevante para obtener predicciones más precisas. Por otra parte, se identificó una correlación importante entre el precio de las acciones y atributos derivados del *análisis técnico bursátil*[27] que comúnmente son utilizados por el inversionista para tomar decisiones basadas en tendencias previas en el mercado de valores. Esto sugiere que los datos históricos de las transacciones en el mercado son útiles para hacer predicciones basadas en patrones cíclicos. En la fase de *modelado y evaluación* se revisaron dos

[26] Con base en metodologías CRISP-DM, KDD y TDSP (Tab, Buck, & Sharkey, 2021) (IBM)
[27] Análisis basado en el supuesto de que los movimientos en el mercado son patrones que se repiten de forma cíclica (Khadjeh Nassirtoussi, Aghabozorgi, Ying Wah, & Ngo, 2014)

modelos de aprendizaje automático; *regresión con redes neuronales artificiales* y *agrupamiento por k medias.* A partir del modelo de *redes neuronales artificiales* se llegó a la conclusión de que la integración de noticias financieras contribuye significativamente a reducir el error en la predicción de los datos; mientras que el modelo de *agrupamiento por k medias* se utilizó para identificar las condiciones del mercado bajo las cuáles es más favorable realizar transacciones de compra y venta de acciones; así como las condiciones en que el mercado presenta mayores valores de volatilidad.

Respecto a los modelos de redes neuronales propuestos para realizar regresión y clasificación se identificó un sobreajuste; a partir de lo que se concluye que los modelos son apropiados siempre y cuando el usuario los entrene frecuentemente; de otra forma, los resultados son poco precisos. Tomando en cuenta que el mercado bursátil reacciona a todos los eventos y situaciones que acontecen alrededor de él, y que los modelos planteados realizan predicciones únicamente para un día posterior a la fecha en que se realiza la consulta, es razonable que sea necesario reentrenar los modelos de forma habitual. La obtención de predicciones precisas es relevante dado que estas pueden representar ganancias o pérdidas en el patrimonio del inversionista; en este sentido, y dado que la herramienta lo permite, es recomendable entrenar el modelo cada vez que se haga la recolección de nuevos datos.

La herramienta desarrollada en este proyecto es una propuesta que hasta donde sabemos, en la literatura actual no existe precedente. Si bien existen herramientas que predicen el comportamiento del precio de instrumentos financieros que se transaccionan en el mercado de valores; ninguna está basada en la metodología CRISP-DM (Jiang, 2021). En este sentido, el presente trabajo aporta un antecedente para la aplicación de metodologías de aprendizaje automático, minería de textos y análisis de sentimientos, para generar conocimiento referente al mercado de valores. Se espera que los resultados sean referencia para estudios posteriores que busquen entender el comportamiento de los instrumentos financieros que se transaccionan en mercado bursátil. No obstante, quedan abiertos diversos puntos que es necesario revisar en trabajos futuros. Es pertinente revisar de forma detallada el efecto de las noticias financieras publicadas durante los días en que el mercado bursátil se encuentra cerrado; así como el efecto de la variación en los parámetros utilizados para el cálculo de atributos derivados y parámetros de los modelos de

redes neuronales, con el propósito de confirmar si juegan un papel relevante en el comportamiento de los modelos. También queda pendiente el traslado de la herramienta desarrollada a un entorno de PaaS, de forma que sea accesible en línea sin necesidad de disponer de diversos sistemas informáticos para hacer uso de ella.

6 Anexos

6.1 Código en Python para la recolección de datos de Google News

```
# Recolectar noticias de las acciones "stocks"
# Adaptación realizada a partir de las siguientes publicaciones:
Hurin el 10/02/2021 en el sitio: https://pypi.org/project/GoogleNews/
Boghe, Kristof el 24/04/2021 en el sitio: https://pypi.org/project/nordvpn-switcher/
##

#Intsalar API GoogleNews
pip install --upgrade GoogleNews

#Importar las librerias y paquetes
import sys
import ssl
from io import StringIO
import urllib.request
import datetime
from datetime import date
import time
import requests
from bs4 import BeautifulSoup as Soup
import time
import csv
import pandas as pd
from nordvpn_switcher import initialize_VPN,rotate_VPN,terminate_VPN

#Las 50 acciones [stocks] más representativas del índice S&P500
stocks=[
'Apple "AAPL"','Microsoft "MSFT"','Amazon "AMZN"','Facebook "FB"','Tesla "TSLA"','Alphabet "GOOGL"',
'Alphabet "GOOG"','Berkshire Hathaway "BRK-B"','Johnson & Johnson "JNJ"','JPMorgan Chase "JPM"','NVIDIA
"NVDA"', 'Visa "V"','Walt Disney "DIS"','PayPal "PYPL"','Procter & Gamble "PG"','UnitedHealth "UNH"','Home
Depot "HD"', 'Mastercard "MA"','Bank of America"BAC"','Netflix "NFLX"','Intel "INTC"','Comcast
"CMCSA"','Adobe "ADBE"', 'Verizon "VZ"','Abbott "ABT"','salesforce "CRM"','Exxon "XOM"','Cisco
"CSCO"','AT&T "T"','Walmart "WMT"', 'Pfizer "PFE"','Thermo Fischer "TMO"','PepsiCo "PEP"','Coca-Cola
"KO"','Broadcom "AVGO"','Merck & Co "MRK"', 'AbbVie "ABBV"','NIKE "NKE"','Chevron "CVX"','Qualcomm
"QCOM"','NextEra Energy "NEE"','Accenture Plc "ACN"', 'McDonald"s "MCD"','Eli Lilly "LLY"','Texas
Instruments "TXN"','Medtronic "MDT"','Costco "COST"', 'Citigroup "C"','Honeywell "HON"','WellsFrago "WFC"']

#Generar lista con los códigos bursátiles [tickers] de cada accion
tickers =[]
for t in range (0,len(stocks)):
    ticker_start = stocks[t].find('"')
    tickers.append(stocks[t][ticker_start+1:len(stocks[t])-1])
print(len(tickers),tickers)

#NiordVPN Proxy Settings
initialize_VPN(save=1,area_input=['complete rotation'])
start_time = time.time()

#Definir periodo en formato (AAAA,MM,DD)
```

```
date = datetime.datetime(2021,1,1)
date_1 = datetime.datetime(2021,1,2)
dias = 2 #cantidad de días a buscar, p.e. 1año --> 365 días

#Obtener datos de Google News
news =[]
from GoogleNews import GoogleNews
googlenews = GoogleNews()
googlenews = GoogleNews(lang='en')
googlenews = GoogleNews(period='d')
googlenews = GoogleNews(encode='utf-8')

for k in range(dias):
    rotate_VPN()
    formatted_date = datetime.date.strftime(date, "%m/%d/%Y")
    formatted_date_1 = datetime.date.strftime(date_1, "%m/%d/%Y")
    googlenews = GoogleNews(start=formatted_date,end=formatted_date)

    for i in range (len(tickers)):
        googlenews.clear()
        old_stdout = sys.stdout
        result = StringIO()
        sys.stdout = result
        googlenews.search(stocks[i])
        sys.stdout = old_stdout
        result_string = result.getvalue()
        print(result_string)

        while result_string == 'HTTP Error 429: Too Many Requests\n':
            print("Rotate VPN")
            rotate_VPN()
            old_stdout = sys.stdout
            result = StringIO()
            sys.stdout = result
            googlenews.search(stocks[i])
            sys.stdout = old_stdout
            result_string = result.getvalue()

        #googlenews.getpage(2)
        #googlenews.getpage(3)
        print(news)
        news.append(googlenews.results())
        googlenews.clear()
        for j in range (len(news[len(news)-1])):
            news[len(news)-1][j]["ticker"]=tickers[i]
        #time.sleep(1)
        print(date,stocks[i])
    #print(date)
    #print(proxy[k])
    date += datetime.timedelta(days=1)
    date_1 += datetime.timedelta(days=1)
terminate_VPN()
elapsed_time = time.time() - start_time
print(elapsed_time)
```

```
#Archivo CSV creado con datos obtenidos de Google News
csv_columns = ['title', 'media','date', 'datetime','desc','link','img','ticker']
csv_file = "news_manual.csv"

try:
   with open(csv_file, 'w', newline='',errors='ignore') as csvfile:
      writer = csv.DictWriter(csvfile, fieldnames=csv_columns)
      writer.writeheader()
      for i in range(len(news)):
         #print(i)
         writer.writerows(news[i])
except IOError:
   print("I/O error")
```

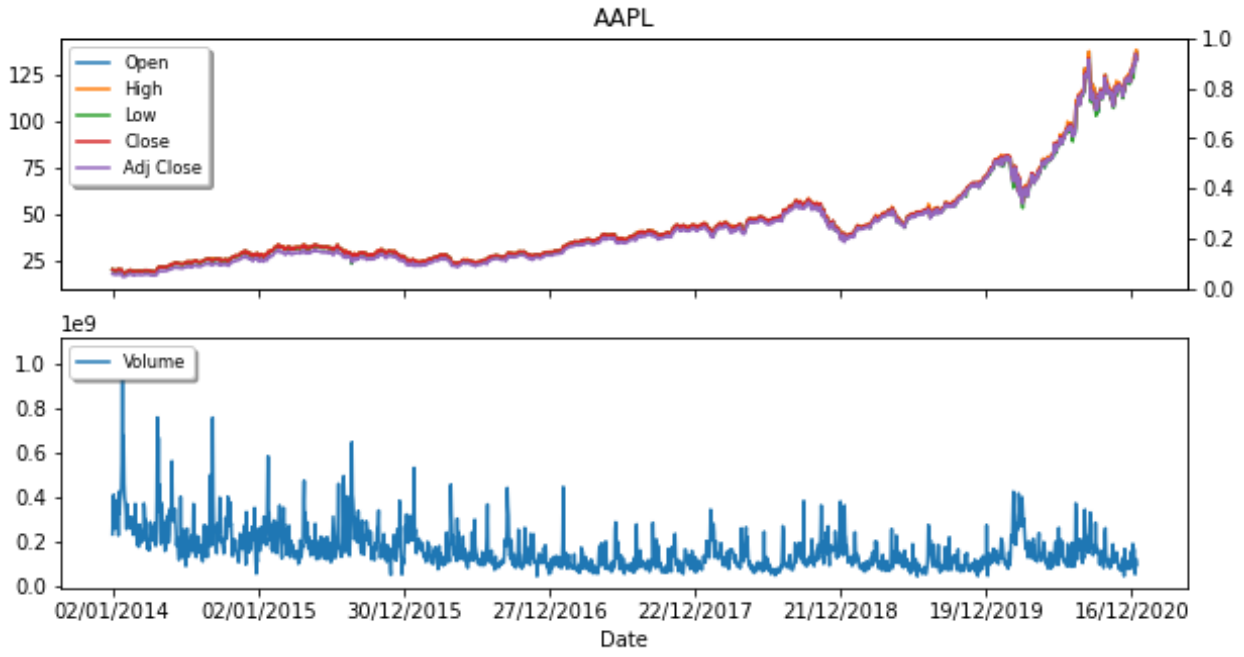
6.2 Código en Python para la recolección de datos de Yahoo Finance

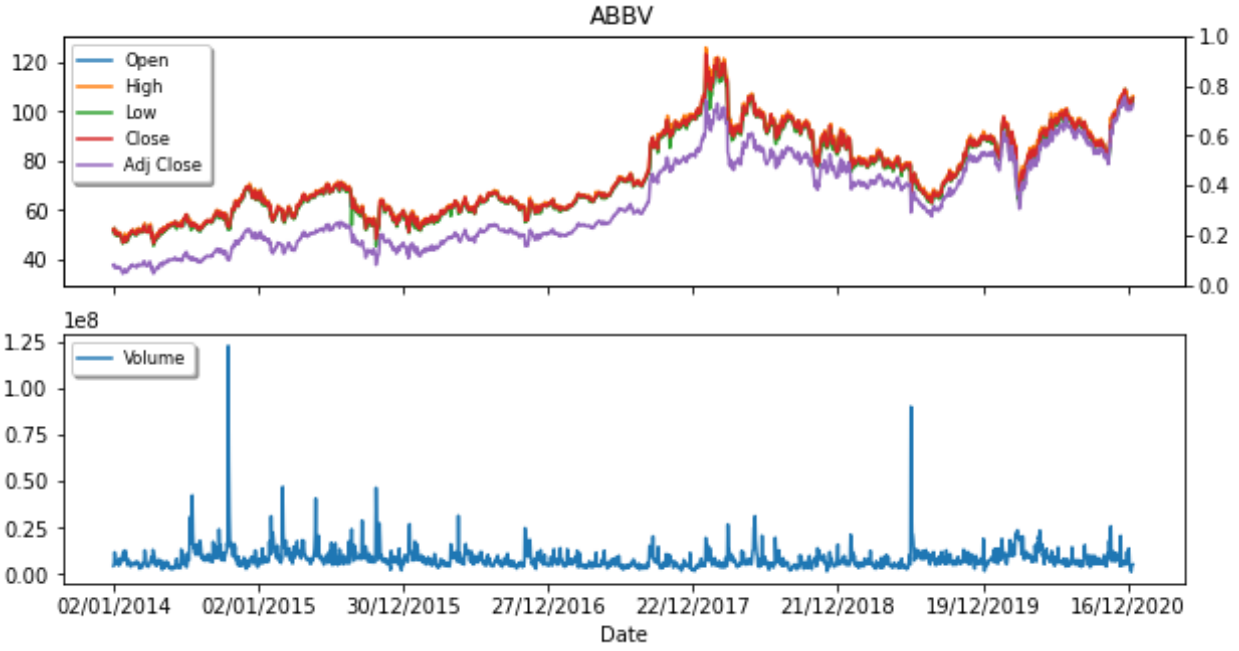
```
# Recolectar datos transaccionales de las acciones, e indicadores económicos
# Adaptación realizada a partir de las siguientes publicaciones:
Aroussi, Ran el 19/10/2021 en el sitio: https://pypi.org/project/yfinance/
Boghe, Kristof el 24/04/2021 en el sitio: https://pypi.org/project/nordvpn-switcher/
##

#Indicadores económicos
import yfinance as yf
import numpy as np
import pandas as pd
from pandas import DataFrame

index =[
'Volatility "^VIX"', 'MSCI_Emerging_Market "EEM"', '10-YR_Treasury_Index "^TNX"', 'Crude_Oil_Apr_21
"CL=F"', 'E-Mini S&P 500 "ES=F"', 'USD/MXN "MXN=X"', 'USD/EUR "EUR=X"', 'Gold "GC=F"']

data =[]

for i in range(len(tickers)):
   data.append(yf.download(tickers[i], start="2015-01-01", end="2020-12-31"))
   data[i]['Ticker'] = tickers[i]


#Recolectar datos de índices económicos de Yahoo Finance
index_tickers =[]

for t in range (0,len(index)):
   index_start = index[t].find('"')
   #print(stocks[t][ticker_start+1:len(stocks[t])])
   index_tickers.append(index[t][index_start+1:len(index[t])-1])
print(index_tickers)
print(len(index_tickers))

indicators =[]
indic =[]
for i in range(len(index_tickers)):
```

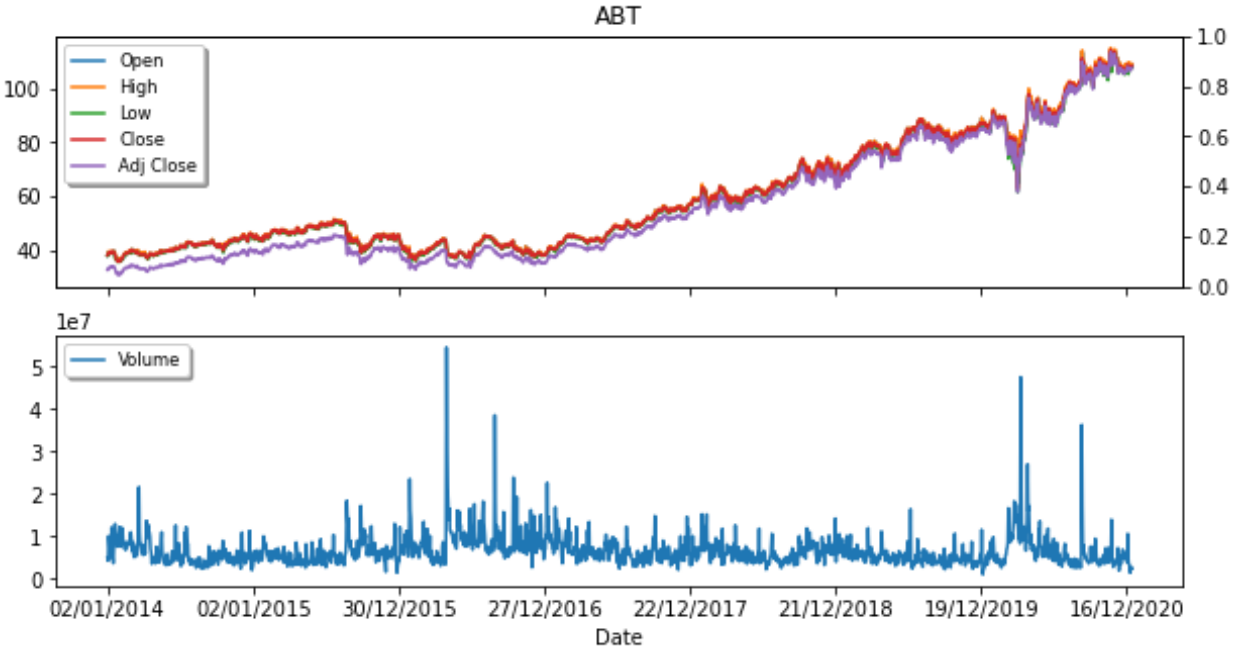
```
    indicators.append(yf.download(index_tickers[i], start="2020-12-30", end="2020-12-31"))
    indicators[i]['Ticker'] = index_tickers[i]


#Archivo CSV creado con datos obtenidos de Yahoo Finance (Datos transaccionales)
filename = "data.csv"
# opening the file with w+ mode truncates the file
f = open(filename, "w+")
f.close()

for j in range(len(data)):
    data_0=pd.DataFrame(data[j])
    if j <= 0:
        data_0.to_csv('data.csv', mode='a',header=True)
    else:
        data_0.to_csv('data.csv', mode='a',header=False)


#Archivo CSV creado con datos obtenidos de Yahoo Finance (Indicadores)
filename = "data_i.csv"
# opening the file with w+ mode truncates the file
f = open(filename, "w+")
f.close()

for j in range(len(indicators)):
    data_i0=pd.DataFrame(indicators[j])
    if j <= 0:
        data_i0.to_csv('data_i.csv', mode='a',header=True)
    else:
        data_i0.to_csv('data_i.csv', mode='a',header=False)
#df_indicators.to_csv('data_i.csv', mode='a',header=True)
```

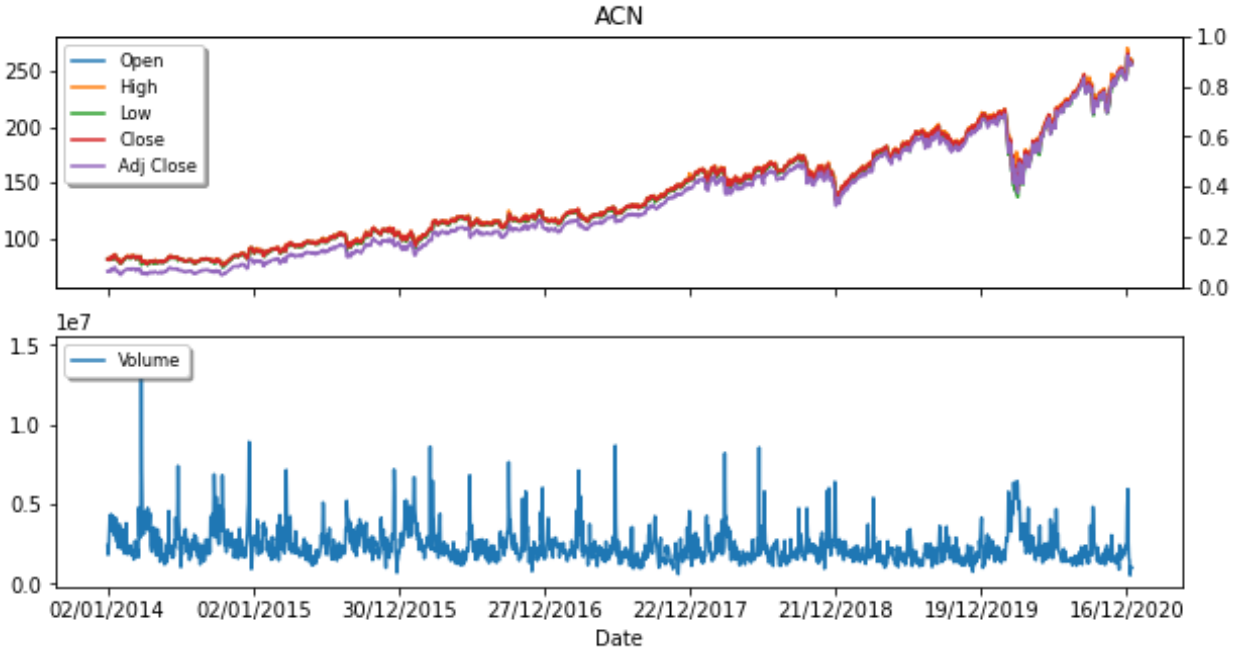
6.3 Gráfica de los datos recolectados de Yahoo Finance

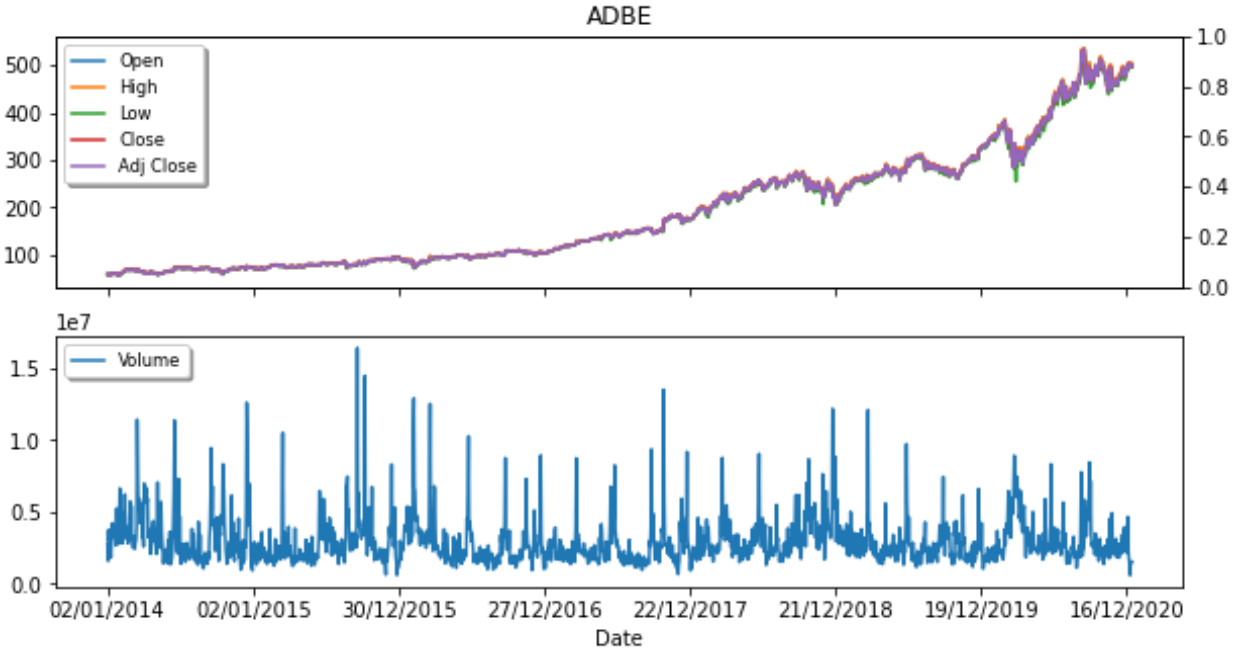
A continuación, se muestran los gráficos del conjunto de datos obtenido de Yahoo Finance correspondiente a las primeras diez acciones del índice bursátil S&P500 de acuerdo con su orden alfabético.

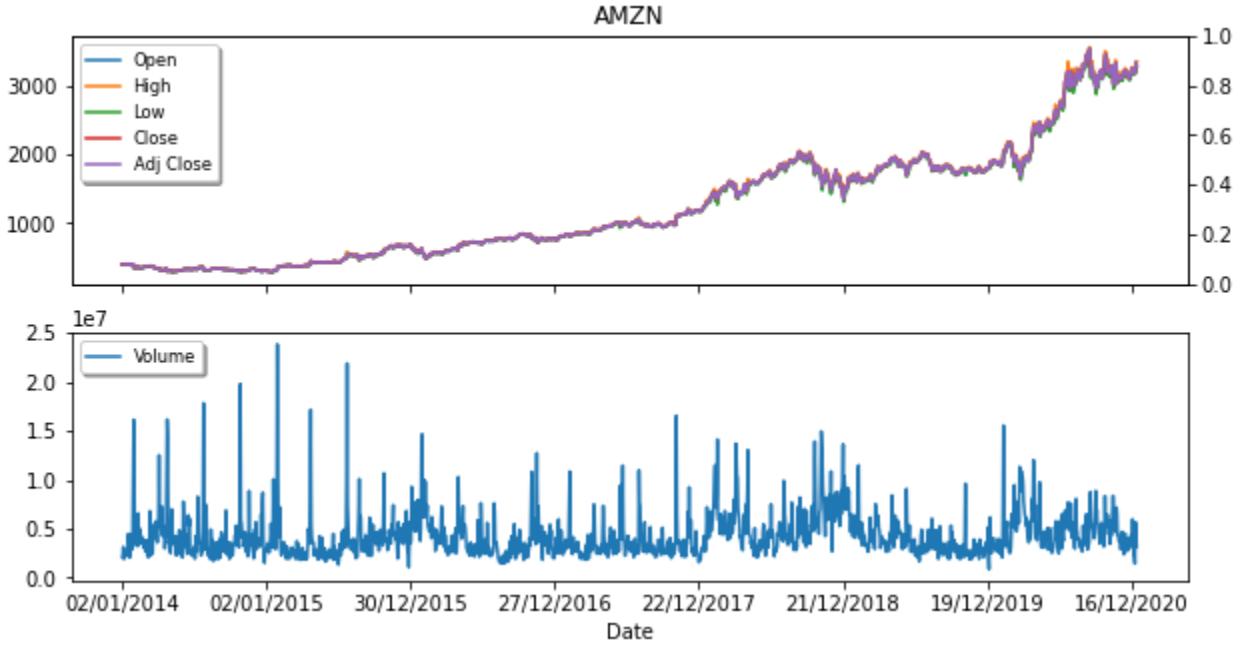


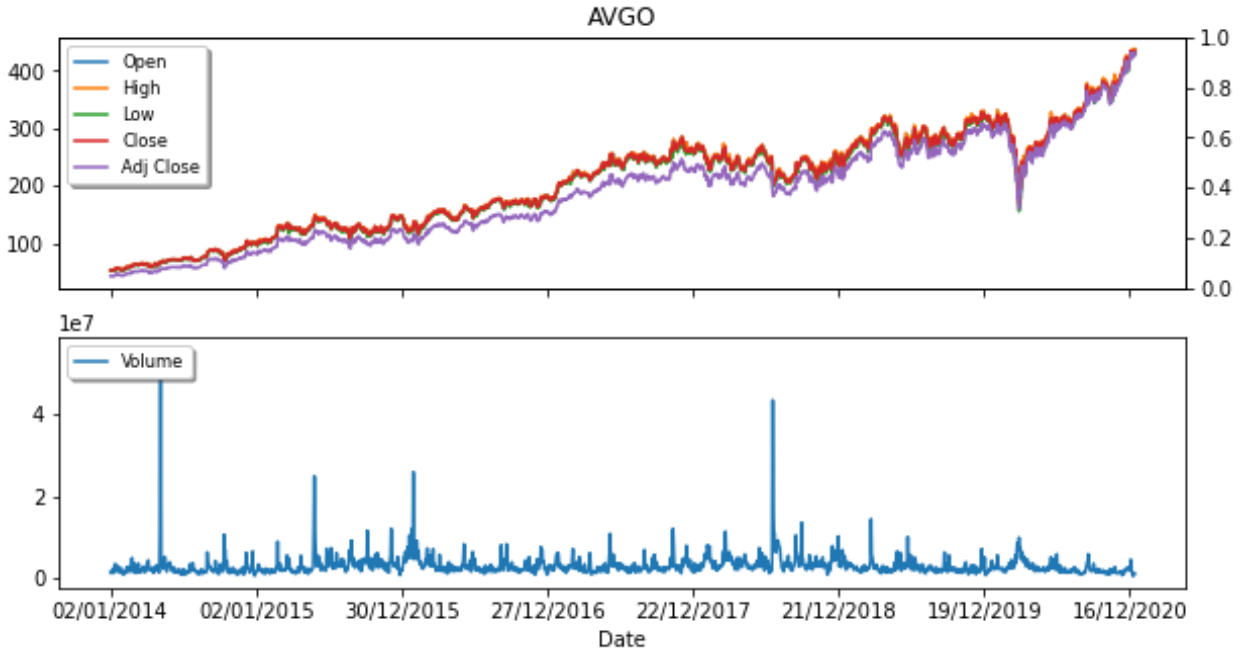

ABBV
Open
High
Low
Close
Adj Close
1e8
Volume
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date

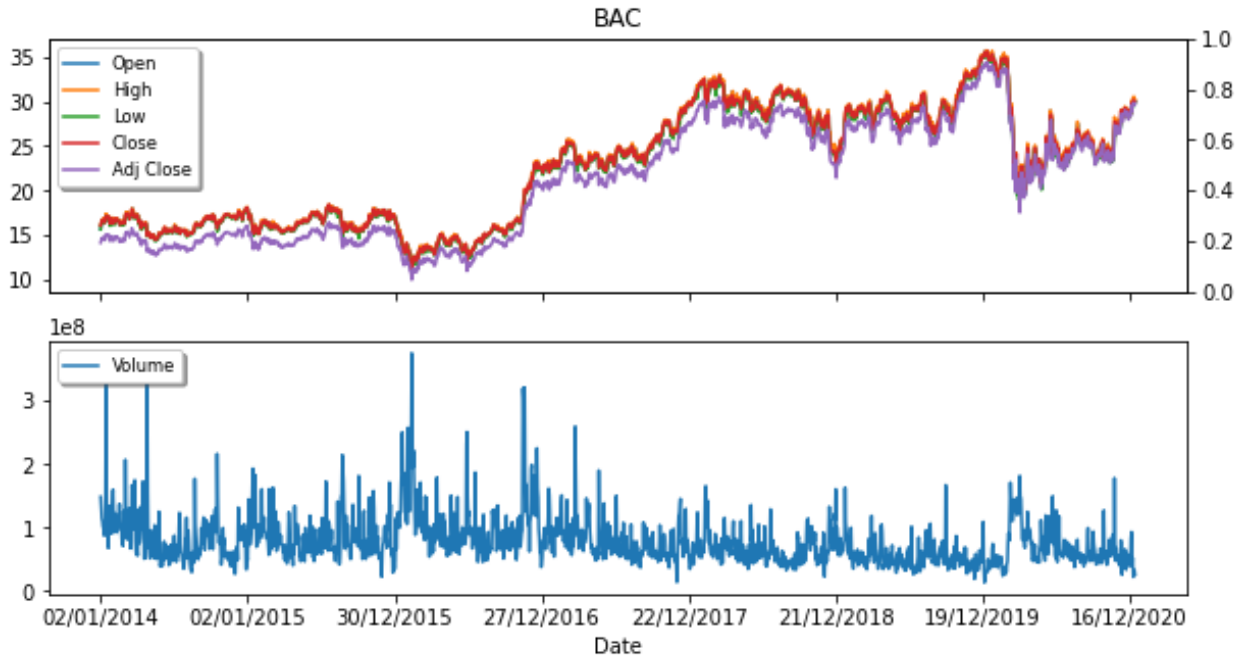

ABT
Open
High
Low
Close
Adj Close
1e7
Volume
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date

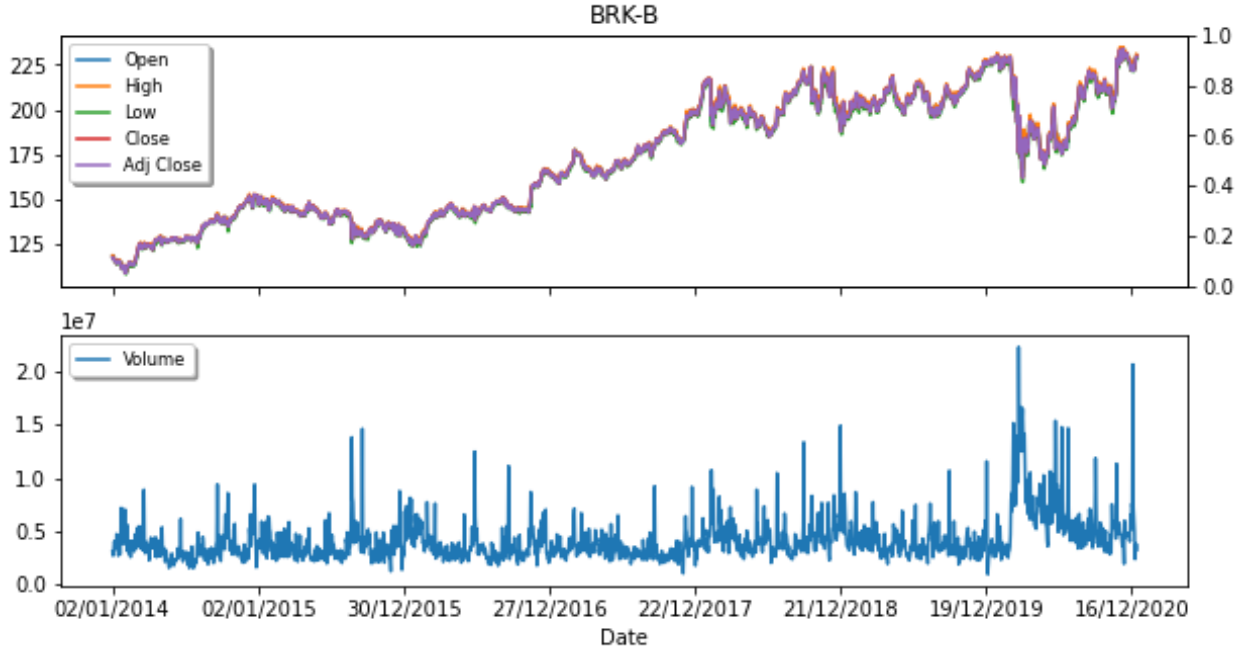

ACN
Open
High
Low
Close
Adj Close
Volume
1e7
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date

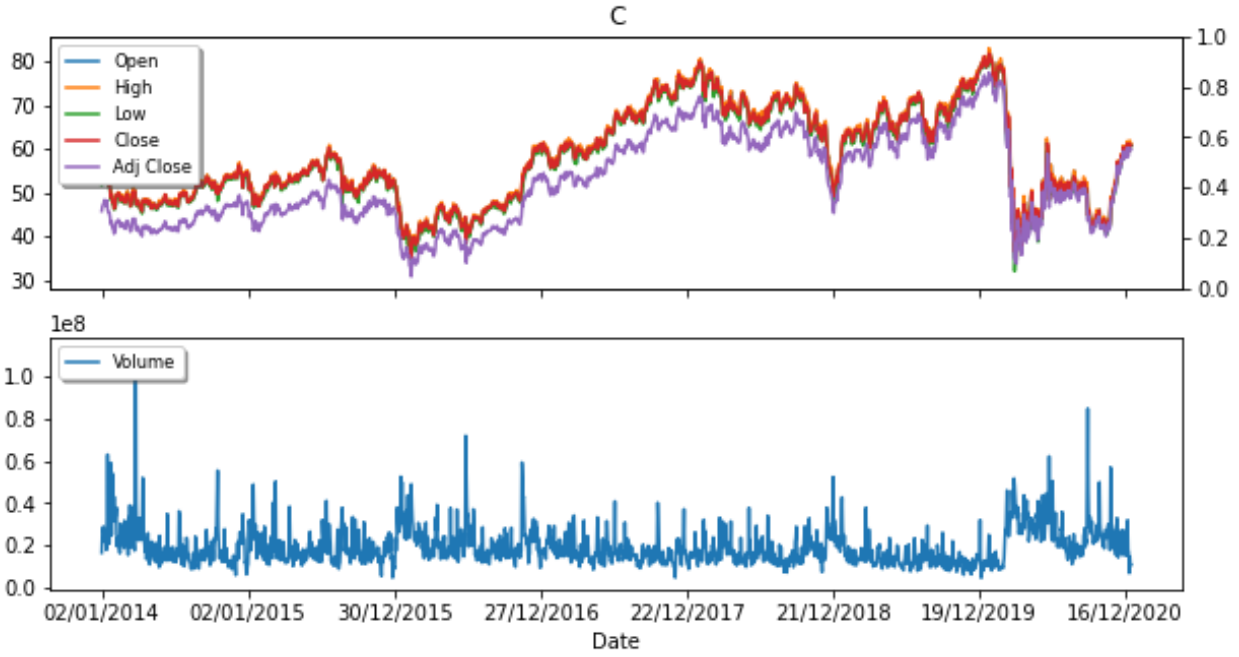

ADBE
Open
High
Low
Close
Adj Close
Volume
1e7
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date


AMZN
Open
High
Low
Close
Adj Close
3000
2000
1000
1.0
0.8
0.6
0.4
0.2
0.0
1e7
Volume
2.5
2.0
1.5
1.0
0.5
0.0
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date


AVGO
Open
High
Low
Close
Adj Close
400
300
200
100
1.0
0.8
0.6
0.4
0.2
0.0
1e7
Volume
4
2
0
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date


BAC
Open
High
Low
Close
Adj Close
1e8
Volume
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date


BRK-B
Open
High
Low
Close
Adj Close
1e7
Volume
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date


C
Open
High
Low
Close
Adj Close
80
70
60
50
40
30
1.0
0.8
0.6
0.4
0.2
0.0
1e8
Volume
1.0
0.8
0.6
0.4
0.2
0.0
02/01/2014
02/01/2015
30/12/2015
27/12/2016
22/12/2017
21/12/2018
19/12/2019
16/12/2020
Date

7 Referencias

Abraham, A. (2005). Artificial Neural Networks. (P. H. Sydenham, & R. Thorn, Eds.) *Handbook of Measuring System Design*, 901-908. doi:https://doi.org/10.1002/0471497398.mm421

Agrawal, M., Khan, A., & Shukla, P. (2019, julio). Stock Price Prediction using Technical Indicators:. *International Journal of Recent Technology and Engineering (IJRTE), 8*(2), 2277-3878. doi:10.35940/ijrteB3048.078219

Alquraan, T., Alqisie, A., & Shorafa, A. A. (2016). Do Behavioral Finance Factors Influence Stock Investment Decisions of Individual Investors? (Evidences from Saudi Stock Market). *American International Journal of Contemporary Research, 6*(3), 159-169. Retrieved from http://www.aijcrnet.com/journals/Vol_6_No_3_June_2016/16.pdf

Alsabban, S., & Alarfaj, O. (2019). An Empirical Analysis of Behavioral Finance in the Saudi Stock Market: Evidence of Overconfidence Behavior. *nternational Journal of Economics and Financial Issues*, 73-86. doi:https://doi.org/10.32479/ijefi.8920

Amadeo, K. (2021, 03 30). *10-Year Treasury Note and How It Works*. Retrieved from The Balance: https://www.thebalance.com/10-year-treasury-note-3305795

Amazon Web Services, I. (2021). *Amazon Machine Learning Developer Guide*. Retrieved from Amazon Web Services: https://docs.aws.amazon.com/machine-learning/latest/dg/model-fit-underfitting-vs-overfitting.html

Amazon Web Services, Inc. (2021). *Amazon Machine Learning Guía para desarrolladores.* Retrieved from Amazon Web Services: https://docs.aws.amazon.com/es_es/machine-learning/latest/dg/cross-validation.html

Ang, A., Goetzmann, W. N., & Schaefer, S. M. (2011). *The Efficient Market Theory and Evidence: Implicatios for Active Investment Management.* Hanover, Massachusetts, EE.UU.: now Publishers, Inc.

Apergis, N., Cooray, A., & Rehman, M. U. (2017). Do Energy Prices Affect U.S. Investor Sentiment? *Journal of Behavioral Finance, 19*(2), 122-140. doi:https://doi.org/10.1080/15427560.2017.1373354

Aroussi, R. (2020, 10 05). *yfinance 0.1.55* . Retrieved from PyPI: https://pypi.org/project/yfinance/

Atkins, A., Niranjan, M., & Gerding, E. (2018). Financial news predicts stock market volatility better than close price. *The Journal of Finance and Data Science, 4*(2), 120-137. doi: https://doi.org/10.1016/j.jfds.2018.02.002

Attigeri, G. V., Manohara, P. M., Pai, R. M., & Nayak, A. (2015). Stock Market Prediction: A Big Data Approach. *TENCON 2015 - 2015 IEEE Region 10 Conference*, 1-5. doi:https://doi.org/10.1109/TENCON.2015.7373006

Ausenbaugh, E., Faller, M., & Cohen, C. (2020, 12 11). *2020's most notable market events, Part I* . Retrieved from J.P. Morgan: https://www.jpmorgan.com/wealth-management/wealth-partners/insights/2020s-most-notable-market-events-part-I

Azevedo, A. I., & Santos, M. F. (2012, 06 04). *KDD, SEMMA and CRISP-DM: a parallel overview*. Retrieved from Instituto Politécnico do Porto. : https://recipp.ipp.pt/handle/10400.22/136

Baker, M., & Wurgler, J. (2007). Investor Sentiment in the Stock Market. *Journal of Economic Perspectives, 21*(2), 129-151. doi:https://doi.org/10.1257/jep.21.2.129

Baker, R. S. (n.d.). *Data Mining for Education.* Retrieved from Carnegie Mellon University: School of Computer Science: http://www.cs.cmu.edu/~rsbaker/Encyclopedia%20Chapter%20Draft%20v10%20-fw.pdf

Baker, S. R., Bloomb, N., Davis, S., & Sammon, M. (2019, noviembre). *What Triggers Stock Market Jumps?* Retrieved from Stockmarket Jumps: https://stockmarketjumps.com/files/BBDS_BigJumps.pdf

Bampinas, G., & Panagiotidis, T. (2017). Oil and stock markets before and after financial crises: A local Gaussian correlation approach. *Journal of Futures Markets, 37*(12), 1179-1204. doi:https://doi.org/10.1002/fut.21860

Banco de México. (2021, 03 15). *Indicadores Diarios de la Bolsa Mexicana de Valores - (CF103)*. Retrieved from Banco de México - Sistema de Información Económica: https://www.banxico.org.mx/SieInternet/consultarDirectorioInternetAction.do?accion=consultarCuadro&idCuadro=CF103&sector=7&locale=es

Berrar, D. (2019). Cross-validation. In *Encyclopedia of Bioinformatics and Computational Biology* (pp. 542-545). Elsevier. doi:10.1016/B978-0-12-809633-8.20349-X

Bhagat, P. (2019, marzo). *Financial Markets-(Basic Concepts).* Retrieved from ResearchGate: https://www.researchgate.net/publication/331501828_Financial_Markets-Basic_Concepts

Bholowalia, P. a. (2014). EBK-means: A clustering technique based on elbow method and k-means in WSN. *International Journal of Computer Applications, 105*(9), 18-19. Retrieved from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.735.7337&rep=rep1&type=pdf

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python.* O'Reilly Media Inc. Retrieved from http://nltk.org/book

BMV. (2019, 03 21). *El Índice de Precios y Cotizaciones y su importancia para el mercado*. Retrieved from Hablemos de Bolsa: https://blog.bmv.com.mx/2019/03/el-indice-de-precios-y-cotizaciones/

Boghe, K. (2021, 04 24). *nordvpn-switcher 0.2.7*. Retrieved from PyPI: https://pypi.org/project/nordvpn-switcher/

Brownlee, J. (2020, 12 11). *Perceptron Algorithm for Classification in Python*. Retrieved from Machine Leaning Mastery: https://machinelearningmastery.com/perceptron-algorithm-for-classification-in-python/

California State University Long Beach. (n.d.). *History of the Perceptron*. Retrieved from California State University Long Beach: https://home.csulb.edu/~cwallis/artificialn/History.htm

Carson, R. (2018, 12 23). *A Look Back At 10 Of The Top Financial News Stories Of 2018*. Retrieved from Forbes: https://www.forbes.com/sites/rcarson/2018/12/23/a-look-back-at-10-of-the-top-financial-news-stories-of-2018/?sh=4b9ed587bd1d

Cboe Exchange, Inc. (2021). *Mini VIX™ Futures Contracts Now Trading*. Retrieved from Cboe Exchange, Inc.: https://www.cboe.com/tradable_products/vix/

Cboe Global Markets. (2021). *U.S. Equities Market Volume Summary*. Retrieved from Cboe Global Markets: https://www.cboe.com/us/equities/market_share/market/2021-03-11/

Chandra, A. (2008). Decision Making in the Stock Market: Incorporating Psychology with Finance. *National Conference on Forecasting Financial Markets of India*. Retrieved from https://ssrn.com/abstract=1501721

Chandradevan, R. (2017, 08 17). *Radial Basis Functions Neural Networks — All we need to know*. Retrieved from Towards Data Science: https://towardsdatascience.com/radial-basis-functions-neural-networks-all-we-need-to-know-9a88cc053448

Choudhry, R., & Garg, K. (2008). A Hybrid Machine Learning System for Stock Market Forecasting. *World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering, 2*(3), 689-692. Retrieved from https://publications.waset.org/8952/a-hybrid-machine-learning-system-for-stock-market-forecasting

Chukwuchekwa Ulumma, J. (2011). Comparing the performance of backpropagation algorithm and genetic algorithms in pattern recognition problems. *International Journal of Computer Information Systems, 2*(5), 7-12. Retrieved from https://www.researchgate.net/profile/Joy_Chukwuchekwa/publication/331843599_Comparing_the_performance_of_Backpropagation_algorithm_and_Genetic_Algorithm/links/5c900447a6fdcc38175cab5a/Comparing-the-performance-of-Backpropagation-algorithm-and-Genetic-Algo

Cowles, A. (1933, julio). Can Stock Market Forecasters Forecast. *Econometrica, 1*(3), 309-324. doi:doi.org/10.2307/1907042

Cruz, M. (2007). La globalización como estrategia de desarrollo: la evidencia de los países desarrollados. *Investigación económica, 66*(259), 103-131. Retrieved from http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-16672007000100103

Cutler, D. M., Poterba, J. M., & Summers, L. H. (1988). What Moves Stock Proces? *National Bureau of Economic Research*(2538), 1-19. Retrieved from https://www.nber.org/system/files/working_papers/w2538/w2538.pdf

Darškuvienė, V. (2010). *Financial Markets.* Vytautas Magnus University. Retrieved from https://www.bcci.bg/projects/latvia/pdf/7_Financial_markets.pdf

Data Science Project Management. (n.d.). *What is CRISP DM?* Retrieved from Data Science Project Management: https://www.datascience-pm.com/crisp-dm-2/

De Long, J., Shleifer, A., Summers, L., & Waldmann, R. (1990). Noise Trader Risk in Financial Markets. *Journal of Political Economy, 98*(4), 703-738. Retrieved from http://www.jstor.org/stable/2937765

DeLancey, J. (2020, 05 29). *Pros and Cons of NLTK Sentiment Analysis with VADER*. Retrieved from CodeProject: https://www.codeproject.com/Articles/5269447/Pros-and-Cons-of-NLTK-Sentiment-Analysis-with-VADE

Desai, R. (2020, 06 08). *The Most Popular Tools and Software for Data Science*. Retrieved from Towards Data Science: https://towardsdatascience.com/top-20-most-popular-tools-for-data-science-93c5618893a4

Dhami, S., al-Nowaihi, A., & Sunstein, C. R. (2019, agosto 20). Heuristics and Public Policy: Decision-making Under Bounded Rationality. *Studies in Microeconomics*, 7-58. doi:https://doi.org/10.1177/2321022219832148

Dorman, J. (2017, 12 31). *5 Big Economic Stories Of 2017 And 5 Things To Watch In 2018*. Retrieved from Forbes: https://www.forbes.com/sites/jeffreydorfman/2017/12/31/5-big-economic-stories-of-2017-and-5-things-to-watch-in-2018/?sh=51fc4d402322

Dowie, G. (2020, 10 21). *MACD and Stochastic: A Double-Cross Strategy*. Retrieved from Investopedia: https://www.investopedia.com/articles/trading/08/macd-stochastic-double-cross.asp

Drew, C. J. (2008). Introduction to Qualitative Research and Mixed-Method Designs. Designing and Conducting Research in Education. *Designing and Conducting Research in Education*, 183-208. doi:https://doi.org/10.4135/9781483385648.n8

Eastwood, B. (2020, 06 18). *The 10 Most Popular Programming Languages to Learn in 2021*. Retrieved from Northeastern University : https://www.northeastern.edu/graduate/blog/most-popular-programming-languages/

Economoua, F., Hassapiscand, C., & Philippa, N. (2018). Investors' fear and herding in the stock market. *Applied Economics, 50*(34-35), 3654-3663. doi:https://doi.org/10.1080/00036846.2018.1436145

Elbahloul, K. (2019). Stock Market Prediction Using Various Statistical Methods. *Statistical and Machine Learning Analysis, 1*, 1-7. Retrieved from https://doi.org/10.13140/rg.2.2.13235.17446

Ergeshidze, A. (2017). Impact of Exchange Rate on Macroeconomic Indicators. *Ilia State University*, 1-7. doi:https://dx.doi.org/10.2139/ssrn.3038890

Fernando, J. (2021, 01 05). *Moving Average Convergence Divergence (MACD)*. Retrieved from Investopedia: https://www.investopedia.com/terms/m/macd.asp

Geladi, P., & Linderholm, J. (2020). Principal Component Analysis. *Comprehensive Chemometrics*, 17-37. doi:10.1016/B978-0-12-409547-2.14892-9

Ghojogh, B., & Crowley, M. (2019). The Theory Behind Overfitting, Cross Validation,. 1-20. Retrieved from https://arxiv.org/pdf/1905.12787.pdf

Goetzmann, W. N., & Kumar, A. (2008). Equity Portfolio Diversification. *Review of Finance, 12*(3), 433-463. doi:https://doi.org/10.1093/rof/rfn005

Google Sites. (2020, 08 30). *k-means clustering algorithm*. Retrieved from Data Clustering Algorithms: https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm

Gopinath, G., Milesi-Ferretti, G. M., & Nabar, M. (2019, 12 18). *2019 in Review: The Global Economy Explained in 5 Charts*. Retrieved from International Monetary Fund: https://blogs.imf.org/2019/12/18/2019-in-review-the-global-economy-explained-in-5-charts/

Gupta, J. N., & Sexton, R. S. (1999). Comparing backpropagation with a genetic algorithm for neural network training. *Omega, 27*(6), 679 - 684. doi:https://doi.org/10.1016/S0305-0483(99)00027-4

Hand, D. J. (2015). Data Mining. *Wiley StatsRef: Statistics Reference Online*, 1-7. doi:10.1002/9781118445112.stat06466.pub2

Harvill, J. (2020). *Evaluating Estimators Bias and Mean Squared Error*. Retrieved from Department of Statistical Science - Baylor University: https://mediaspace.baylor.edu/media/Evaluating+Point+EstimatorsA+Bias+and+Mean+Squared+Error/1_hkw077wz/170953002

Hawley, D. D., Johnson, J. D., & Raina, D. (1990). Artificial Neural Systems: A New Tool for Financial Decision-Maling. *Financial Analyst Journal, 46*(6), 63-72. doi:doi.org/10.2469/faj.v46.n6.63

Hayes, A. (2021, junio 25). *Stochastic Oscillator*. Retrieved from Investopedia: https://www.investopedia.com/terms/s/stochasticoscillator.asp

Heyes, A. (2020, 12 03). *Stochastic Oscillator*. Retrieved from Investopedia: https://www.investopedia.com/terms/s/stochasticoscillator.asp

Hu, H. (2021, 02 10). *GoogleNews 1.5.5* . Retrieved from PyPI: https://pypi.org/project/GoogleNews/

Hu, X., Tang, J., Gao, H., & Liu, H. (2013). Unsupervised sentiment analysis with emotional signals. *WWW '13: Proceedings of the 22nd international conference on World Wide Web*, 607-618. doi:https://doi.org/10.1145/2488388.2488442

Huber, S., Wiemer, H., Schneider, D., & Ihlenfeldt, S. (2019). DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP-DM model. *79*, 403-408. doi:https://doi.org/10.1016/j.procir.2019.02.106

Huguenard, J. (2017). *Error Sum of Squares (SSE)*. Retrieved from Stanford UNiversity: https://hlab.stanford.edu/brian/error_sum_of_squares.html

IBM Analytics. (2015). *Metodología Fundamental para la Ciencia de Datos.* Retrieved from IBM: https://www.ibm.com/downloads/cas/6RZMKDN8

IBM. (n.d.). *Conceptos básicos de ayuda de CRISP-DM. IBM.* Retrieved from IBM Knowledge Center: https://www.ibm.com/support/knowledgecenter/es/SS3RA7_sub/modeler_crispdm_ddita/clementine/crisp_help/crisp_overview.html

Indurkhya, N., & Damerau, F. J. (2010). *Handbook of Natural Language Processing.* Boca Raton, Florida, EE.UU.: Chapman & Hall/CRC.

Jiang, W. (2021, diciembre). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications, 184*, 1-22. doi:https://doi.org/10.1016/j.eswa.2021.115537

Jiawei, Y. (2019). Automated Sentiment Analysis of Text Data with NLTK. *Journal of Physics: Conference Serie, 1187*(5), 1-8. doi:https://doi.org/10.1088/1742-6596/1187/5/052020

Johnson, D. (2021, marzo 12). *What is a CSV file? How to open, use, and save the popular spreadsheet file in 3 different apps*. Retrieved from Business Insider: https://www.businessinsider.com/what-is-csv-file?r=MX&IR=T

Jones, C. I. (2018, 11 18). To Close the Gap. *Science, 334*(6058), 906-906. doi:https://doi.org/10.1126/science.1214409

Jordan, J. (2017, 07 21). *Evaluating a machine learning model*. Retrieved from Jeremy Jordan: https://www.jeremyjordan.me/evaluating-a-machine-learning-model/

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends,perspectives, and prospects. *Science, 349*(6245), 255-260. doi:10.1126/science.aaa8415

K, M. T., S., N., & R., S. U. (2013). Stock Market Reaction to Political Events (Evidence from Pakistan). *Journal of Economics and Sustainable Development , 4*(1), 165-174. Retrieved from https://core.ac.uk/download/pdf/234645816.pdf

Kayalar, D. E., Küçüközmen, C. C., & Selcuk-Kestel, A. s. (2017). The impact of crude oil prices on financial market indicators: copula approach. *Energy Economics, 61*, 162-173. doi:https://doi.org/10.1016/j.eneco.2016.11.016

Khadjeh Nassirtoussi, A., Aghabozorgi, S., Ying Wah, T., & Ngo, D. C. (2014). Text mining for market prediction: A systematic review. *Expert Systems with Applications, 41*(16), 7653-7670. doi:https://doi.org/10.1016/j.eswa.2014.06.009

Khan, K., & Sahai, A. (2012). A Comparison of BA, GA, PSO, BP and LM for Training Feed forward Neural Networks in e-Learning Context. *I.J. Intelligent Systems and Applications, 7*, 23-29. doi:10.5815/ijisa.2012.07.03

Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H., & Alfakeeh, A. S. (2020, marzo 14). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Compu*, 1-24. doi:10.1007/s12652-020-01839-w

Khan, W., Ghazanfar, M., Azam, M., Karami, A., Alyoubi, K., & Alfakeeh, A. (2020). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, 1-24. doi:10.1007/s12652-020-01839-w

King, R. G., Plosser, C. I., & Rebelo, S. T. (1988). Production, Growth and Business Cycles. *Journal of Monetary Economics, 21*, 195-232. Retrieved from http://people.bu.edu/rking/EC702/kprjme88a.pdf

Kingdom of Saudi Arabia. (2018). *Financial Investments and Stock Markets.* Capital Market Authority. Retrieved from http://www.arbahcapital.com/sites/default/files/2018-05/publication%20%284%29.pdf

Kogid, M., Asid, R., Lily, J., Mulok, D., & Loganathan, N. (2012, Kogid, Mori & Asid, Rozilee & Lily, Jaratin & Mulok, Dullah & Loganathan, Nanthakumar). The Effect of Exchange Rateson Economic Growth: Empirical Testing on Nominal Versus Real. *The IUP Journal of Financial Economics, 10*(1), 7-17. Retrieved from https://www.researchgate.net/publication/231233782_The_Effect_of_Exchange_Rates_on_Economic_Growth_Empirical_Testing_on_Nominal_Versus_Real_The_Effect_of_Exchange_Rates_on_Economic_Growth_Empirical_Testing_on_Nominal_Versus_Real

Kohara, K., Ishikawa, T., Fukuhara, Y., & Nakamura, Y. (1997). Stock Price PredictionUsing Prior Knowledge and Neural Networks. *International Journal of Intelligent Systems in Accounting, Finance & Management, 6*(1), 11-22. doi:https://doi.org/10.1002/(SICI)1099-1174(199703)6:1<11::AID-ISAF115>3.0.CO;2-3

Krogh, A. (2008). What are artificial neural networks? *Nature Biotechnology, 26*(2), 195-197. doi:https://doi.org/10.1038/nbt1386

Kröse, B., & Van der Smagt, P. (1996, noviembre). *An Introduction to Neural Networks.* Amsterdam: The University of Amsterda,. Retrieved from https://www.infor.uva.es/~teodoro/neuro-intro.pdf

Krotov, V., & Silva, L. (2018). Legality and Ethics of Web Scraping. *Twenty-fourth Americas Conference on Information Systems*, 1-5. Retrieved from https://www.researchgate.net/profile/Vlad-Krotov/publication/324907302_Legality_and_Ethics_of_Web_Scraping/links/5aea622345851588dd8287dc/Legality-and-Ethics-of-Web-Scraping.pdf

Kudryavtsev, A., Cohen, G., & Hon-Snir, S. (2013). Rational" or "Intuitive": Are Behavioral BiasesCorrelated Across Stock Market Investors? *ContemporaryEconomics, 7*(2), 31-53. Retrieved from https://www.econstor.eu/bitstream/10419/105372/1/755764218.pdf

Kumar, P. (2017, 10 21). *An Introduction to N-grams: What Are They and Why Do We Need Them?* Retrieved from XRDS: The ACM Magazine for Students: https://blog.xrds.acm.org/2017/10/introduction-n-grams-need/

Laker, K. (2006, 05). *Benefits of a Multi-Dimensional Model*. Retrieved from Oracle Corporation : https://www.oracle.com/technetwork/developer-tools/warehouse/benefits.pdf

Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., . . . Deng, X. (2013). Empirical analysis: Stock Market Prediction Via Extreme Learning Machine. *Neural Computing and Applications, 27*(1), 67-78. doi:https://doi.org/10.1007/s00521-014-1550-z

López-Cabarcos, M. Á., Pérez-Pico, A., Vázquez-Rodríguez, P., & López-Pérez, M. L. (2019). Investor sentiment in the theoretical field of behavioural finance. *Economic Research-Ekonomska Istraživanja, 33*(1), 2101-2119. doi:https://doi.org/10.1080/1331677x.

Mahmood, H. (2019, 01 02). *Gradient Descent*. Retrieved 2021, from Towards Data Science: https://towardsdatascience.com/gradient-descent-3a7db7520711

Malkiel, B. G. (1989). Efficient Market Hypothesis. In M. M. Eatwell J., *Finance* (pp. 127-134). London: Palgrave Macmillan. doi:https://doi.org/10.1007/978-1-349-20213-3_13

Mallawaarachchi, V. (2017, 07 07). *Introduction to Genetic Algorithms — Including Example Code*. Retrieved from Towards Data Science: https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3

Mateu Gordon, J. L. (n.d.). *MACD*. Retrieved from Expansión - Diccionario económico: https://www.expansion.com/diccionario-economico/macd.html

Matsubara, Y. (2021, 01 069). *PyMySQL 1.0.2* . Retrieved from PyPi: https://pypi.org/project/PyMySQL/

Microsoft. (2021, septiembre 20). *Windows commands*. Retrieved from Microsoft Ignite: https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/windows-commands

Miller, J. I., & Ratti, R. A. (2009). Crude oil and stock markets: Stability, instability, and bubbles. *Energy Economics, 31*(4), 559-568. doi:https://doi.org/10.1016/j.eneco.2009.01.009

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of Machine Learning.* Londres, Inglaterra: Massachusetts Institute of Technology.

Moreno Iglesias, A. (2020, 07 20). *Data normalization with Pandas and Scikit-Learn*. Retrieved from Towards Data Science: https://towardsdatascience.com/data-normalization-with-pandas-and-scikit-learn-7c1cc6ed6475

MSCI Inc. (2021). *Emerging Markets*. Retrieved from MSCI Inc.: https://www.msci.com/our-solutions/index/emerging-markets

Najeb M. H., M. (2013). The Impact of Stock Market Performance upon Economic Growth. *International Journal of Economics and Financial Issues, 3*(4), 788-798. Retrieved from https://econjournals.com/index.php/ijefi/article/download/557/pdf

Ngoc, L. T. (2014). Behavior Pattern of Individual Investors in Stock Market. *International Journal of Business and Management, 9*(1), 1-16. doi:10.5539/ijbm.v9n1p1

NLTK Project. (2020, 08 13). *Natural Language Toolkit*. Retrieved from NLTK Project: https://www.nltk.org/

Norani, N., Shareduwan, M., & Kasihmuddin, M. A. (2021). Logic Learning in Adaline Neural Network. *Pertanika Journals: Science & Technology, 29*(1), 285-300. Retrieved from http://www.pertanika.upm.edu.my/resources/files/Pertanika%20PAPERS/JST%20Vol.%2029%20(1)%20Jan.%202021/16%20JST-2143-2020.pdf

NordVPN. (2021). *NordVPN*. Retrieved from NordVPN: https://nordvpn.com/es/

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2019). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review, 53*(4), 3007-3057. doi:https://doi.org/10.1007/s10462-019-09754-z

Önder, Z., & Şimga-Muğan, C. (2006). How Do Political and Economic News Affect Emerging Markets? Evidence from Argentina and Turkey. *Emerging Markets Finance & Trade*, 50-77. Retrieved from https://www.jstor.org/stable/27750507

Oracle. (2021). *MySQL Workbench*. Retrieved from MySQL: https://www.mysql.com/products/workbench/

Oracle Corporation. (2003, 12). *2 The Multidimensional Data Model*. Retrieved from Oracle OLAP: Application Developer's Guide: https://docs.oracle.com/cd/B13789_01/olap.101/b10333/multimodel.htm

Örkcü, H. H., & Bal, H. (2011). Comparing performances of backpropagation and genetic algorithmsin the data classification. *Expert Systems with Applications, 38*(4), 3703 - 3709. doi:https://doi.org/10.1016/j.eswa.2010.09.028

Ousterhout, & K., J. (1998, marzo). *Scripting: Higher-Level Programming*. Retrieved from Stanford University: https://web.stanford.edu/~ouster/cgi-bin/papers/scripting.pdf

Pandarachalil, R., Sendhilkumar, S., & Mahalakshmi, G. (2015). Twitter Sentiment Analysis for Large-Scale Data: An Unsupervised Approach. *Cognitive Computation, 7*, 254-262. doi:https://doi.org/10.1007/s12559-014-9310-z

Pedersen, T. B. (2009). Multidimensional Modeling. In T. B. Pedersen, & L. Liu (Ed.), *Encyclopedia of Database Systems* (pp. 1777-1784). Boston, MA, USA: Springer US. doi:https://doi.org/10.1007/978-0-387-39940-9_229

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12*, 2825-2830. Retrieved from https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf?source=post_page----------

Petrusheva, N., & Jordanoski, I. (2016). Comparative analysis between the fundamental and technical analysis of stocks. *Journal of Process Management. New Technologies, 4*(2), 26-31. doi:https://doi.org/10.5937/jpmnt1602026p

Piech, C. (2013). *K Means*. Retrieved from Stanford University: https://stanford.edu/~cpiech/cs221/handouts/kmeans.html

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics, 3*(2), 143-157. doi:https://doi.org/10.1016/j.joi.2009.01.003

Pražák, T. (2018). The Effect of Economic Factors on Performance of the Stock Market in the Czech Republic. *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 66*(6), 1613-1626. doi:https://doi.org/10.11118/actaun201866061613

Press, G. (2016, marzo 23). *Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says*. Retrieved from Forbes: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/?sh=1684a9c96f63

Princeton University. (n.d.). *WorNet: A Lexical database for English*. Retrieved from Princeton University: https://wordnet.princeton.edu

Project Jupyter. (2021, 02 08). *Jupyter*. Retrieved from Project Jupyter: https://jupyter.org/

Project Jupyter Revision. (2021). *ipywidgets*. Retrieved from Jupyter Widgets: https://ipywidgets.readthedocs.io/en/latest/

Python. (2021, 04). *ssl — TLS/SSL wrapper for socket objects*. Retrieved from Python: https://docs.python.org/es/3.8/library/ssl.html

Python. (2021, 04). *sys — Parámetros y funciones específicos del sistema*. Retrieved from Python: https://docs.python.org/es/3.10/library/sys.html

Python. (2021, 04). *urllib.request — Extensible library for opening* . Retrieved from Python: https://docs.python.org/3/library/urllib.request.html

Redman, T. C. (2018, 04 03). *If Your Data Is Bad, Your Machine Learning Tools Are Useless.* Retrieved from Harvard Business Review: https://hbr.org/2018/04/if-your-data-is-bad-your-machine-learning-tools-are-useless

Refaeilzadeh, P., Tang, L., & Liu, H. (2016). Cross-Validation. In L. Liu, & M. T. Özsu (Eds.), *Encyclopedia of Database Systems* (pp. 1-7). New York: Springer. doi:10.1007/978-1-4899-7993-3_565-2

Reilly, F. K., & Brown, K. C. (2011). *Investment Analysis and Portfolio Management* (10 ed.). Cengage Learning.

Renault, T. (2020). Sentiment Analysis and Machine Learning in Finance: A Comparison of Methods and Models on One Million Messages. *Digital Finance, 2*, 1-13. doi:https://doi.org/10.1007/s42521-019-00014-x

Ritter, J. R. (2003). Behavioral finance. *Pacific-Basin Finance Journal, 11*(4), 429-437. doi:10.1016/s0927-538x(03)00048-9

Ross, S. (2020, 10 01). *How Do S&P 500 Futures Work?* Retrieved from Investopedia: https://www.investopedia.com/ask/answers/042315/how-do-sp-500-futures-work.asp

S&P 500. (2021). Retrieved from S&P Global: https://www.spglobal.com/spdji/en/indices/equity/sp-500/#overview

S&P Dow Jones Indices. (2021). *S&P 500*. Retrieved from S&P Dow Jones Indices: https://www.spglobal.com/spdji/en/indices/equity/sp-500/#overvie

Schmeling, M. (2007). *An empirical analysis of behavioral finance theories in international equity markets.* Retrieved from Deutsche National Bibliothek: https://d-nb.info/987036203/34

Schumaker, R. P., & Chen, H. (2009). Textual analysis of stock market prediction using breaking financial news. *ACM Transactions on Information Systems, 27*(2), 1-19. doi:https://doi.org/10.1145/1462198.1462204

Science Direct. (2021). *Extreme Learning Machine*. Retrieved from Science Direct: https://www.sciencedirect.com/topics/engineering/extreme-learning-machine

Shanhong, L. (2020, 12 22). *Ranking of the most popular database management systems worldwide, as of December 2020*. Retrieved from Statista:

https://www.statista.com/statistics/809750/worldwide-popularity-ranking-database-management-systems/

Shavlik, J. W., & Dietterich, T. G. (1990). *Readings in Machine Learning.* (M. B. Morgan, Ed.) San Mateo, California, EE.UU.: Morgan Kaufmann Publishers, Inc.

Shen, S., Jiang, H., & Zhang, T. (2012). Stock Market Forecasting Using Machine Learning Algorithms.

Siddique, M., & Tokhi, M. (2001). Training Neural Networks: Backpropagation vs Genetic Algorithms. *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222), 4*, 2673-2678. doi:10.1109/IJCNN.2001.938792

Sindhu, M. I., Hussain Bukhari, S. M., & Hussain, A. (2017). Macroeconomic Factors do influencing Stock Price: A Case Study on Karachi Stock Exchange. *Journal of Economics and Sustainable Development, 5*(7), 114-124. Retrieved from https://core.ac.uk/download/pdf/2346463

Singh, V., Piryani, R., Uddin, A., Waila, P., & Marisha. (2013). Sentiment analysis of textual reviews; Evaluating machine learning, unsupervised and SentiWordNet approaches. *2013 5th International Conference on Knowledge and Smart Technology (KST)*, 122-127. doi:10.1109/KST.2013.6512800

Songhao, W. (2020, 05 23). *3 Best metrics to evaluate Regression Model?* Retrieved from Towards Data Science: https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b

Spradlin, D. (2019, 08 23). *Are You Solving the Right Problem?* . Retrieved from Harvard Business Review: https://hbr.org/2012/09/are-you-solving-the-right-problem

Springboard India. (2020, 08 31). *Best language for Machine Learning: Which Programming Language to Learn*. Retrieved from Springboard: https://in.springboard.com/blog/best-language-for-machine-learning/

SQL Alchemy. (2021). *The Python SQL Toolkit and Object Relational Mapper*. Retrieved from SQL Alchemy: https://www.sqlalchemy.org/

SSL Support Team. (2019, 10 02). *What Is SSL?* Retrieved from SSL: https://www.ssl.com/faqs/faq-what-is-ssl/

State Street Global Advisors. (2021, 03 11). *SPDR S&P 500 ETF Trust*. Retrieved from State Street Global Advisors: https://www.ssga.com/us/en/institutional/etfs/funds/spdr-sp-500-etf-trust-spy

StockCharts. (n.d.). *MACD(Moving Average Convergence Divergence Oscillator)*. Retrieved from StockCharts: https://school.stockcharts.com/doku.php?id=technical_indicators:moving_average_convergence_divergence_macd

Subash, R. (2012, 06 28). *Role of Behavioral Finance in Portfolio Investment Decisions: Evidence from India.* Retrieved from Univerzita Karlova, Fakulta sociálních věd: https://dspace.cuni.cz/handle/20.500.11956/43150

Sullivan, J. (2016, 12 30). *The Biggest Financial Headlines of 2016*. Retrieved from US News: https://money.usnews.com/investing/articles/2016-12-30/year-in-review-the-biggest-financial-headlines-of-2016

Tab, M., Buck, A., & Sharkey, K. (2021, noviembre 12). *The Team Data Science Process lifecycle*. Retrieved from Microsoft: https://docs.microsoft.com/en-us/azure/architecture/data-science-process/lifecycle

The MathWorks, Inc. (2021). *Calling MATLAB from Python*. Retrieved from Help Center | Mathworks: https://www.mathworks.com/help/matlab/matlab-engine-for-python.html?s_tid=CRUX_lftnav

The Mathworks, Inc. (2021). *Matlab*. Retrieved from Mathworks: https://www.mathworks.com/products/matlab.html

Trevino, A. (2016, 12 06). *Introduction to K-means Clustering*. Retrieved from Oracle AI and Data Science Blog: https://blogs.oracle.com/datascience/introduction-to-k-means-clustering

Tsang, W., & Chong, T. (2009). Profitability of the On-Balance Volume Indicator. *Economics Bulletin, 29*(3), 2424-2431. Retrieved from www.accessecon.com/pubs/eb/2009/volume29/eb-09-v29-i3-p87.pdf

Tufféry, S. (2011). *Data Mining and Statistics for Decision Making.* Rennes, Francia: John Wiley & Sons.

University of Cincinnati. (2018, 10 02). *Creating text features with bag-of-words, n-grams, parts-of-speach and more*. Retrieved from UC Business Analytics R Programming Guide: http://uc-r.github.io/creating-text-features

University of Edinburgh. (2020, septiembre 04). *VPN (Virtual Private Network)*. Retrieved from Information Services: https://www.ed.ac.uk/information-services/computing/desktop-personal/vpn

Vargas, M., dos Anjos, C., Bichara, G., & Evsukoff, A. (2018). Deep Leaming for Stock Market Prediction Using Technical Indicators and Financial News Articles. *2018 International Joint Conference on Neural Networks (IJCNN)*, 1-8. doi:10.1109/IJCNN.2018.8489208

Ved, M. (. (2018, 07 20). *Feature Selection and Feature Extraction in Machine Learning: An Overview.* Retrieved from Medium: https://medium.com/@mehulved1503/feature-selection-and-feature-extraction-in-machine-learning-an-overview-57891c595e96

Vessiari, K. (2016, 01 07). *Timeline: The Major Economic Events of 2015*. Retrieved from Orbex: https://www.orbex.com/blog/en/2016/01/timeline

Virlics, A. (2013). Investment Decision Making and Risk. *Procedia Economics and Finance, 6*, 169-177. doi:https://doi.org/10.1016/s2212-5671(13)00129-9

Voila-Gallery. (2021). *Voila*. Retrieved from Voila: https://voila-gallery.org/

Voskoglou, C. (2017, 05 05). *What is the best programming language for Machine Learning?* Retrieved from Towards Data Science: https://towardsdatascience.com/what-is-the-best-programming-language-for-machine-learning-a745c156d6b7

Walker, A. (2014, 12 23). *The world economy in 2014*. Retrieved from BBC: https://www.bbc.com/news/business-30400861

White, H. (1988). Economic Prediction Using Neural Networks: The Case of IBM Daily Stock Returns. *IEEE 1988 International Conference on Neural Networks, 2*, 451-458. doi:10.1109/ICNN.1988.23959

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for DataMining. In P. A. Company, & N. Mackin (Ed.), *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining* (pp. 29-39). Blackpool, Lancashire : Practical Application Company . Retrieved from http://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf

Yahoo. (n.d.). *Exchanges and data providers on Yahoo Finance*. Retrieved from Yahoo: https://help.yahoo.com/kb/exchanges-data-providers-yahoo-finance-sln2310.html

Yahoo Finance. (2021, 03 11). *SPDR S&P 500 ETF Trust (SPY)*. Retrieved from Yahoo Finance: https://finance.yahoo.com/quote/SPY?p=SPY&.tsrc=fin-srch

Yalçın, O. G. (2020, 12 12). *Sentiment Analysis in 10 Minutes with Rule-Based VADER and NLTK*. Retrieved from Towards Data Science: https://towardsdatascience.com/sentiment-analysis-in-10-minutes-with-rule-based-vader-and-nltk-72067970fb71

Yiu, T. (2019, 07 20). *The Curse of Dimensionality: Why High Dimensional Data Can Be So Troublesome*. Retrieved from Towards Data Science: https://towardsdatascience.com/the-curse-of-dimensionality-50dc6e49aa1e

Yoo, P. D., Kim, M. H., & Jan, T. (2005). Machine Learning Techniques and Use of Event Information for Stock Market Prediction: A Survey and Evaluation. *International Conference on Computational Intelligence for Modelling, Control and Automation and*

International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), 835-841. doi:10.1109/CIMCA.2005.1631572

Zapata Claveria, M. A. (2015, abril 01). Expertos y libertad en el paternalismo libertario. *Revista Digital Universitaria - UNAM, 16*(4), 1-1. Retrieved from http://www.revista.unam.mx/vol.16/num4/art27/

Zhai, Y., Hsu, A., & Halgamuge, S. (2007). Combining News and Technical Indicators in Daily Stock Price Trends Prediction. *Advances in Neural Networks – ISNN 2007*, 1087-1096. doi:10.1007/978-3-540-72395-0_132

Zhao, B. (2017). Web Scraping. *Encyclopedia of BigData*, 1-3. doi:10.1007/978-3-319-32001-4_483-1

Zhen-Guo, C., Tzu-An, C., & Zhen-Hua, C. (2011). Feed-forward neural networks training: A comparison between genetic algorithm and back-propagation learning algorithm. *International Journal of InnovativeComputing, Information and Control, 7*(10), 5839-5850. Retrieved from http://hdl.handle.net/11536/14780