Instituto Tecnológico y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Matemáticas y Física

Maestría en Ciencia de Datos

Predicción de cambio precio en mercados bursátiles mediante el uso de modelado predictivo para Series de Tiempo

---

TRABAJO RECEPCIONAL que para obtener el **GRADO** de
Maestro en Ciencia de Datos

Presenta:
Carlos Caloca Gómez

Director:
Mtro. Juan Francisco Muñoz Elguezábal

Tlaquepaque, Jalisco, 17 de junio de 2024

Predicción de cambio precio en mercados bursátiles mediante el uso de modelado predictivo para Series de Tiempo

Carlos Caloca Gómez

Abstract

En el contexto actual de los mercados bursátiles, la capacidad de predicción a tiempo y con precisión de las fluctuaciones de precios en los distintos activos es de suma importancia para la toma de decisiones. Este proyecto se enfoca en el desarrollo de un sistema de modelado predictivo utilizando técnicas de ciencia de datos para pronosticar los movimientos del precio. Se utilizan datos históricos de velas de transacciones de criptomoneda Ether, los registros son por hora y el periodo de tiempo es desde enero 2023 hasta octubre 2023. El objetivo del trabajo es desarrollar un proceso de modelado predictivo capaz de estimar variaciones en los precios para determinar el cambio de signo de la variable objetivo. La metodología propuesta incluye la evaluación de modelos de clasificación como perceptrón multicapa y regresión logística, ajuste de hiperparámetros para mejorar la precisión de los modelos. Se utilizan métricas como exactitud, precisión, sensibilidad, valor F1 y área bajo la curva ROC para comparar los resultados de las predicciones. Se busca que este proyecto contribuya a la aplicación de técnicas de ciencia de datos en el campo financiero.

Palabras Clave: Ciencia de Datos, Mercado Bursátil, Modelado Predictivo, Criptomoneda, Análisis Exploratorio de Datos, Perceptrón Multicapa, Regresión Logística.

Tabla de Contenidos

Índice de figuras

Página

Índice de tablas

Página

Dedicatoria

Este trabajo está dedicado a mis seres queridos, por el apoyo constante que contribuyó en mi camino académico y profesional.

A mis maestros y mentores, quienes me enseñaron, guiaron e inspiraron a siempre superar los obstáculos, alcanzar mis objetivos y sobre todo a ir más allá de mis límites. Sin lugar a dudas su orientación, conocimientos y paciencia se ven reflejado en este proyecto.

Este trabajo es para ustedes como testimonio de esfuerzo, superación y gratitud.

1 Introducción

En este capítulo se presenta el contexto del objeto de estudio, justificación, definición del problema y los objetivos del proyecto. Se divide en dos secciones: Importancia y Desafíos de la Predicción del Cambio de Precio y Metodología y Organización del Trabajo.

Este trabajo analiza la capacidad de los modelos predictivos para pronosticar el cambio de precio de la criptomoneda Ether. Siendo importante esto para el desarrollo de estrategias financieras que obtengan resultados exitosos al hacer operaciones bursátiles.

La contribución de este proyecto se enfoca en el análisis de distintos modelos para predecir el cambio de precio utilizando datos históricos, mismos que son utilizados comúnmente en el Análisis Técnico para operaciones bursátiles.

1.1 Importancia y Desafíos de la Predicción del Cambio de Precio

En el mercado bursátil, la capacidad de predecir con precisión las fluctuaciones de precios de los activos es necesaria para la toma de decisiones. Con los avances digitales y la disponibilidad y almacenamiento de información, la ciencia de datos es cada vez más común en este tipo de mercados. El poder identificar tendencias, analizar comportamientos y patrones, así como el desarrollo de modelos predictivos, son tareas que la ciencia de datos puede ayudar a realizar.

La información disponible hoy en día sobre operaciones bursátiles puede ser observada desde un segundo hasta días, incluso semanas y meses. Tal cantidad de información representa un desafío para el Análisis Técnico volviéndolo complejo para realizar. Es por eso que nace la necesidad de utilizar nuevas formas de análisis que ayuden a utilizar grandes cantidades de datos en menor tiempo, reconocimiento de tendencias a un nivel muy pequeño como puede ser utilizar los datos de cada segundo así como la identificación de patrones de comportamiento de las operaciones.

1.2 *Metodología y Organización del Trabajo*

El trabajo se centra en el uso de modelado predictivo con aprendizaje supervisado, específicamente Perceptrón Multicapa y Regresión Logística. Los modelos son utilizados por la necesidad de obtener una respuesta binaria y evaluar la capacidad predictiva del cambio de signo.

El proyecto se divide en 12 capítulos: el capítulo dos presenta el contexto relacionado a este trabajo, el tercero describe el problema a resolver, el cuatro menciona los objetivos a alcanzar, el quinto presenta una revisión de literatura relacionada al tema que se aborda, el sexto describe la obtención de los datos y la metodología utilizada, el séptimo explica el prototipo propuesto, el octavo muestra los resultados obtenidos, el capítulo noveno habla sobre el trabajo futuro, el décimo comparte sesgo e implicaciones del proyecto y los últimos dos capítulos son complementos para el entendimiento de la ejecución del trabajo.

2 Contexto

En este capítulo se presentan aspectos de la industria y contextuales donde sucede el problema. Se divide en tres secciones: Situación Actual, Necesidad de Pronósticos Precisos y Características y Restricciones de la Solución.

2.1 Antecedentes y Motivación

El mercado de criptomonedas ha tenido un crecimiento significativo en los últimos años. Este mercado se caracteriza por su volatilidad y dinamismo. Comúnmente hay 2 enfoques tradicionales para predecir el cambio de precio, análisis técnico y análisis fundamental. Estos se basan en análisis de precios históricos, noticias e indicadores técnicos.

Hoy en día, el acceso a la información, la capacidad de almacenaje y el crecimiento tecnológico hacen que el aprendizaje automático sea cada vez más común. Es por ello que la aplicación de Machine Learning para capturar la complejidad y variabilidad de este mercado es cada vez más común.

Por último, parte del por qué de este proyecto es meramente por el interés en el sector FinTech y su aplicabilidad.

2.2 Situación Actual

El *trading* es una actividad que es cada vez más común, desde el 2019 al 2023 México tuvo un incremento del 65 % de personas que empezaron a hacer esta actividad. Esto se traduce en más de 350 mil personas hasta el 2023 de acuerdo a cifras de la Asociación de Mercados Financieros de América Latina [1].

[1] E. H. Digital, "México se presenta como la futura capital del trading en latinoamérica," 2023. https://heraldodemexico.com.mx/nacional/2023/9/22/mexico-se-presenta-como-la-html [Accessed: (03/05/2024)]

De acuerdo a datos de la Bolsa Mexicana de Valores, se pronostica un aumento del 40 % de *nuevos traders* en los próximos 3 años. México se perfila como la futura capital del *trading* en América Latina [2].

[2] E. H. Digital, "México se presenta como la futura capital del trading en latinoamérica," 2023. https://heraldodemexico.com.mx/nacional/2023/9/22/mexico-se-presenta-como-la-html [Accessed: (03/05/2024)]

2.3 Necesidad de Pronósticos Precisos

María de Jesús Torres Barón, en su trabajo *Pronósticos, una herramienta clave para la planeación de empresas*[3], habla sobre la incertidumbre que convive cualquier empresa o persona para tomar una decisión de negocio. La necesidad de los pronósticos nace para eliminar la intuición de la toma de decisiones.

El proceso para llevar a cabo un pronóstico, como lo sugiere María de Jesús Torres Barón es: formular el problema, recolectar los datos, manipular y limpiar los datos, construir y evaluar el modelo, aplicar el modelo y evaluar el pronóstico [4]. Con esto, se busca reducir la incertidumbre y que la toma de decisiones tenga un respaldo cuantitativo teniendo una visión costo-beneficio.

[3] M. de Jesús Torres Barrón, "Pronósticos, una herramienta clave para la planeación de las empresas," *Instituto Tecnológico de Sonora*, 2022

2.4 El panorama evolutivo de la ciencia estadística y su aplicación en el análisis bursátil

David Spiegelhalter, en su libro *The Art of Statistics*, habla sobre la estrecha relación que hay entre la ciencia estadística con el hecho de producir algoritmos que pudieran ayudar a la toma de decisiones [5]. Y es que para ser específico ante el problema de grandes cantidades de información y con la variación inherente en las actividades bursátiles, es necesario utilizar herramientas que se puedan adaptar y dar certeza para decidir si se compra o vende.

[4] M. de Jesús Torres Barrón, "Pronósticos, una herramienta clave para la planeación de las empresas," *Instituto Tecnológico de Sonora*, 2022

Spiegelhalter menciona que dependiendo del contexto profesional de distintos investigadores, los estadistas recomiendan modelos de regresión, mientras que los científicos en cómputo prefieren la lógica basada en reglas o *red neuronal* [6].

[5] D. Spiegelhalter, *The Art of Statistics How to Learn from Data*. Basic Books, 1st ed., March 2021

Ahora bien, entrando en el contexto bursátil, comúnmente los profesionales toman decisiones bajo su criterio, que pudiera estar influenciado por noticias, intuición u otros análisis [7]. Agregando a esto, Marcos López menciona en su libro *Advances in Financial Machine Learning* que es mucho más fácil encontrar *alphas* (métrica para conocer si una estrategia genera rendimientos por encima o por debajo de lo esperado) en la microestructura que en la macroestructura, es decir, hay mucha oportunidad solamente se requieren herramientas para localizar esas pequeñas oportunidades que a simple vista son difíciles de encontrar [8].

[6] D. Spiegelhalter, *The Art of Statistics How to Learn from Data*. Basic Books, 1st ed., March 2021

[7] M. López de Prado, *Advances in Financial Machine Learning*. John Wiley Sons, Inc., 1st ed., February 2018

Desde una perspectiva técnica, un humano puede identificar patrones en 3 dimensiones mientras que un algoritmo de *Machine Learning* lo hace en 100. Sin embargo, estas herramientas tienen limitantes, como es la falta de explicación teórica a los problemas que resuelven. Esto significa que requieren de un humano para interpretarlo; cuando se obtienen los resultados se necesita de una mente humana para entender e interpretar así como dar una explicación sobre las características que son predictores del fenómeno estudiado [9].

[8] M. López de Prado, *Advances in Financial Machine Learning*. John Wiley Sons, Inc., 1st ed., February 2018

[9] M. López de Prado, *Advances in Financial Machine Learning*. John Wiley Sons, Inc., 1st ed., February 2018

Existen ciertas tendencias, tal y como Stefan Jansen lo aborda en su libro *Machine Learning for Algorithmic Trading* [10], que han acelerado el uso de algoritmos en las operaciones bursátiles para la creación de estrategias y que desde la perspectiva del autor, cambiarán el estilo de como se crean las estrategias. Estas tendencias son las siguientes:

[10] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020

- Crecimiento de disponibilidad de información digital
- Incremento de capacidad computacional y bajos costos para almacenar información
- Avances en métodos estadísticos para grandes conjuntos de datos

Algunos de los usos de *Machine Learning* en operaciones bursátiles que hoy en día existen [11] son:

[11] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020

- Minería de datos para la identificación de patrones y extracción de características
- Generación de factores de riesgo a partir del aprendizaje supervisado
- Prueba y evaluación de estrategias

2.5 *Características y Restricciones de la Solución*

La solución a esta necesidad debe cumplir con distintas características y restricciones. Debe ser capaz de manejar grandes volúmenes de datos históricos, además de identificar patrones, tendencias y relaciones no lineales en los datos. Debe ser capaz de adaptarse a los cambios influenciados por factores económicos, políticos y de regulaciones.

3 Problema

En este capítulo se presenta el problema que este trabajo tiene como enfoque.

En lo que va del año 2024, la criptomoneda Etherium ha experimentado un incremento del 36.44 % (este valor es considerando un tiempo de 4 meses). Sin embargo, considerando solamente el último mes, ha perdido el 5.97 % de su valor. Si se observa el periodo de 1 semana a perdido 0.59 %, pero si vemos el periodo de 1 día, aumentó su valor 0.30 % [1]. Esto nos dice que el precio cambia de manera acelerada y es necesario analizar de manera específica el comportamiento del precio. Si se busca pronosticar el cambio de precio, se vuelve una tarea compleja.

El cambio de precio está infuenciado por distintos factores económicos, sociales y geográficos que no son parte del interés de este trabajo, sino desarrollar un proceso que considere las diferencias del precio para poder pronósticar el cambio de este.

[1] T. View, "Ethusd," 2024. https://es.tradingview.com/symbols/ETHUSD/ [Accessed: (03/05/2024)]

4 Objetivos

En este capítulo se presentan los objetivos del trabajo. Se divide en dos secciones: Objetivo General y Objetivos Específicos.

4.1 Objetivo General

Desarrollar un proceso de modelado predictivo para construir un predictor para el cambio de precio utilizando datos con una frecuencia intradía, con el propósito de pronósticar movimientos en el mercado mediante el cambio de signo de la función:

$$\hat{Y} = C_{t-1} - O_t \tag{4.1}$$

donde $\hat{Y}$ es el resultado de la diferencia entre el precio de cierre en el tiempo t-1 y el precio de apertura en el tiempo t.

4.2 Objetivos Específicos

- Descargar y preprocesar datos históricos del precio intradía para el instrumento ETH/USD de la plataforma Binance Vision.
- Diseñar e implementar un modelo de Perceptrón Multicapa, con más de una sola capa, con consideraciones de regularización y tasa de aprendizaje.
- Diseñar e implementar un modelo de Regresión Logística con consideraciones de regularización.
- Optimizar los modelos mediante el uso de GridSearchCV.

5 Revisión de Literatura

En este capítulo se presenta literatura académica, principalmente trabajos relacionados a este. Se divide en cuatro secciones: Situación Actual en la Predicción de Cambios de Precio en Mercados Financieros, Necesidad de Pronósticos Precisos en un Entorno Volátil, Características y Restricciones de la Soluciiones en la Predicción de Cambios de Precio y Machine Learning aplicadas en Finanzas.

5.1 Situación Actual en la Predicción de Cambios de Precio en Mercados Financieros

Antonio Parisi, en su estudio llamado "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA"[1] presenta distintas ideas que refuerzan la aplicación de la predicción de cambio de precio. En su trabajo expone que en estudios recientes, estos sugieren que las estrategias de transacción basadas en la dirección del cambio de precios son más efectivas que aquellas basadas en una predicción puntual del precio en instrumentos financieros.

[1] A. P. F., "Evaluación de modelos de redes neuronales de predicción del signo de la varaición del ipsa," *Universidad de Chile*, 2002

Antonio Parisi también expone que dado que los mercados financieros se comportan de una manera no lineal, esto ha ayudado al uso de modelos de redes neuronales ya que hay evidencia de buen desempeño de estos modelos en situaciones no lineales.

5.2 Necesidad de Pronósticos Precisos en un Entorno Volátil

Alejo Márquez[2], en su trabajo fin de máster, expone una descripción que cada vez es más relevante, en ella habla sobre la disponibilidad de datos hoy en día. Menciona que hoy hay más datos que ayer pero menos que el día de mañana, el desarrollo tecnológico ha hecho posible el aumento exponencial de los datos. Esto como justificación del cambio de herramientas, donde anteriormente la estadística tradicional era útil, hoy en día existen diversas herramientas, principalmente modelos de inteligencia artifical, para satisfacer las necesidades de estudio.

[2] A. M. Larrea, "Aplicación de técnicas cuantitativas en el análisis de cotizaciones en mercados energéticos," 2022

También, Alejo Márquez[3], menciona que la complejidad radica en las características de cada campo, si estas características pueden perdurar a lo largo del tiempo no será igual a aquellas características que dependen de factores externos como económicos, sociales y políticos. Para caso de mercados bursátiles, el grado de votalidad es alto y hacer estimaciones es arriesgado, esto aumenta dependiendo del alcance del análisis ya sea medio o largo plazo. De la misma

[3] A. M. Larrea, "Aplicación de técnicas cuantitativas en el análisis de cotizaciones en mercados energéticos," 2022

manera afectan a los pronósticos que pueden ser corto, mediano y largo plazo.

5.3 Características y Restricciones de las Soluciones en la Predicción de Cambios de Precio

En el trabajo "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA"por Antonio Parisi [4], el autor expone que el estudio de eventos de transacciones reveló que la omisión de varaibles como rezagos pueden conducir a proyecciones erróneas y resultados poco significativos.

[4] A. P. F., "Evaluación de modelos de redes neuronales de predicción del signo de la varaición del ipsa," *Universidad de Chile*, 2002

Sin embargo, el mismo autor expresa que existen ciertas limitaciones al momento de utilizar modelos de redes neuronales. Estos modelos son difíciles de explicar por su compleja función, no poseen estadísticas clásicas y por ello no es posible realizar pruebas de hipótesis e intervalos de confianza. No existe una guía para seleccionar la dimensión de la red.

5.4 Machine Learning aplicadas en Finanzas

Tal y como lo menciona De Prado [5], los algoritmos de Machine Learning se han convertido en algo común y son utilizados para ejecutar distintas tareas que alguna vez sólo eran hechas por un grupo limitado de personas. Asimismo, De Prado expone que hoy en día las finanzas y Machine Learning van a transformar como las personas invierten.

[5] M. López de Prado, *Advances in Financial Machine Learning*. John Wiley Sons, Inc., 1st ed., February 2018

En la actualidad existen distintas tendencias en la industria de las inversiones que han contribuido a la necesidad de trabajar y desarrollar la conexión de esta industria con Machine Learning. Stefan Jansen [6] menciona que los cambios en la microestructura de mercado, el desarrollo de estrategias de inversión en términos de exposición a factores de riesgo y el constante desarrollo de recursos informáticos han contribuido a que este tema sea más común y más solicitado.

[6] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020

6 Métodos y Datos

En este capítulo se presentan los Métodos y Datos del trabajo. Se divide en dos secciones y cada una tiene distintas subsecciones. La primera sección es Datos y está dividida en: Obtención de datos, Descripción de datos, Estadística Descriptiva, Descripción de los Modelos, Descripción de las Métricas, Análisis Exploratorio, Pre-procesamiento general de datos, Tratamiento de datos faltantes. La segunda es Metodología y está dividida en: Hipótesis, Diseño de experimento, Realización de experimento, Resultados y Conclusión.

6.1 Datos

Binance es un *Exchange* o bien un mercado digital que permite a usuarios comprar o vender criptomonedas. Este mercado, además de permitir la compra/venta, también permite realizar operaciones bursátiles a los usuarios. Adicional a esto, Binance cuenta con un portal llamado **Binance Vision** donde cualquier usuario puede acceder y consultar información de más de 350 criptomonedas. Esta información esta disponible en *https://data.binance.vision*. La información está dividida en 3 mercados: Spot, Margen y Futuros, para este proyecto se utilizó la información del mercado Spot.[1]

[1] Binance, "Binance faq," 2024. https://www.binance.com/es-MX [Accessed: (26/03/2024)]

6.1.1 Obtención de datos

Para descargar la información es necesario seguir los siguientes pasos:

1. Acceder a Binance Vision (https://data.binance.vision)
2. Dar click en el mercado **Spot**
3. Dar click en **daily**
4. Dar click en **klines**
5. Dar click en **ETHTUSD**
6. Dar click en **1h**

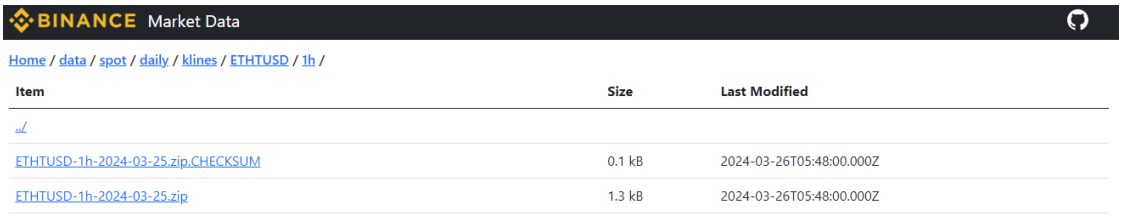

Figura 6.1: Visualización Binance Vision

La Figura 6.1 muestra la visualización del portal después de realizar los 6 pasos del Proceso de Descarga. Es importante recalcar que el formato de cada línea es primero la criptomoneda que seleccionamos, después está presente el tiempo que seleccionamos y por último la fecha en formato año-mes-día. Para la obtención de datos seleccionaremos la opción sin **CHECKSUM**.

Se deben descargar los archivos desde el 2023-01-01 hasta 2023-10-31. Al momento de descargar los archivos, se abren día por día y se copia y pega toda la información en un archivo nuevo. Es importante recalcar que al descargar la información, esta no tiene encabezados. El nombre de los encabezados se puede consultar en esta dirección https://github.com/binance/binance-public-data/. El orden de encabezados es el siguiente:

1 Open Time	2 Open	3 High	4 Low
5 Close+	6 Volume	7 Close Time	8 Quote Asset Volume
9 Number of Trades	10 Taker Buy Base Asset Volume	11 Taker Buy Base Quote Asset Volume	12 Ignore

Tabla 6.1: Tabla demostrativa de los Encabezados.

Por último, en el archivo nuevo de excel, se agrega una nueva columna. Esta columna debe contener únicamente 0 o 1. El criterio para obtener el 0 o 1 es la diferencia de precios de cierre $C_{t-1} - C_t$. Para obtener ese resultado se agrega una condicional en la que si la diferencia de precio es mayor a 0 debemos obtener un 1 o de lo contrario es 0.

6.2 *Descripción de los Datos*

En este apartado se explican las distintas características disponibles en el conjunto de datos utilizado.

- **Open Time:** registro de tiempo de apertura representado en cada hora
- **Open:** precio de apertura en el periodo
- **High:** precio más alto alcanzado en el periodo
- **Low:** precio más bajo alcanzado en el periodo
- **Close:** precio de cierre en el periodo
- **Volume:** volumen total intercambiado en el periodo de tiempo
- **Close Time:** registro de tiempo de cierre representado en cada hora
- **Quote Asset Volume:** volumen total del activo (ETH) en el periodo de tiempo
- **Number of Trades:** número total de operaciones realizadas en el periodo de tiempo
- **Taker Buy Base Asset Volume:** volumen total del activo base comprado durante el periodo de tiempo, en este caso es ETH
- **Taker Buy Quote Asset Volume:** volumen total del activo cotizada comprada durante el periodo de tiempo, en este caso es USD
- **Ignore:** dato que no es relevante para el análisis y puede ser ignorado

Estas son las características básicas que se tienen en el conjunto de datos. La variable de respuesta no está originalmente en el conjunto de datos, esta variable se crea como se mencionó en el punto anterior.

	1672531200000	1195.67	1196.21	1192.21	1193.53	5479.854	1672534799999	6544285.047596	6158	2076.7402	2479779.030506	0	0.1
0	1672534800000	1193.53	1195.73	1193.41	1195.38	2738.3979	1672538399999	3.271612e+06	3308	1254.0049	1.498065e+06	0	1
1	1672538400000	1195.38	1196.10	1193.51	1194.86	2429.3965	1672541999999	2.902240e+06	3194	979.2334	1.169702e+06	0	0
2	1672542000000	1194.85	1194.85	1191.21	1193.47	5148.8046	1672545599999	6.141005e+06	4962	1744.5012	2.080476e+06	0	0
3	1672545600000	1193.47	1193.47	1190.22	1192.34	4333.7378	1672549199999	5.165866e+06	4161	1664.9285	1.984479e+06	0	0
4	1672549200000	1192.34	1193.99	1192.25	1193.87	1544.9423	1672552799999	1.843229e+06	2125	731.0198	8.721530e+05	0	1

Figura 6.2: DataFrame Sin Encabezados o Nombres de las Características Originales

Tal y como se muestra en la Figura 6.2, así es como se ve el dataframe sin ninguna modificación, únicamente es al cargar los datos.

	Open_time	Open	High	Low	Close	Volume	Close_time	Quote_asset_volume	Number_of_trades	Taker_buy_base_asset_volume	Taker_buy_quote_asset_volume	Ignore	Response
0	1672534800000	1193.53	1195.73	1193.41	1195.38	2738.3979	1672538399999	3.271612e+06	3308	1254.0049	1.498065e+06	0	1
1	1672538400000	1195.38	1196.10	1193.51	1194.86	2429.3965	1672541999999	2.902240e+06	3194	979.2334	1.169702e+06	0	0
2	1672542000000	1194.85	1194.85	1191.21	1193.47	5148.8046	1672545599999	6.141005e+06	4962	1744.5012	2.080476e+06	0	0
3	1672545600000	1193.47	1193.47	1190.22	1192.34	4333.7378	1672549199999	5.165866e+06	4161	1664.9285	1.984479e+06	0	0
4	1672549200000	1192.34	1193.99	1192.25	1193.87	1544.9423	1672552799999	1.843229e+06	2125	731.0198	8.721530e+05	0	1

Figura 6.3: DataFrame Con Encabezados o Nombres de las Características Originales

Ahora bien, al agregar los encabezados que previamente fueron explicados, el dataframe se ve de esta manera como aparece en la Figura 6.3.

Una vez que el DataFrame contenga la información suficiente, se procede a utilizar la función *info* para conocer el tipo de variables que se tienen.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7294 entries, 0 to 7293
Data columns (total 13 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Open_time                     7294 non-null   int64
 1   Open                          7294 non-null   float64
 2   High                          7294 non-null   float64
 3   Low                           7294 non-null   float64
 4   Close                         7294 non-null   float64
 5   Volume                        7294 non-null   float64
 6   Close_time                    7294 non-null   int64
 7   Quote_asset_volume            7294 non-null   float64
 8   Number_of_trades              7294 non-null   int64
 9   Taker_buy_base_asset_volume   7294 non-null   float64
 10  Taker_buy_quote_asset_volume  7294 non-null   float64
 11  Ignore                        7294 non-null   int64
 12  Response                      7294 non-null   int64
dtypes: float64(8), int64(5)
memory usage: 740.9 KB
```

Figura 6.4: Información de las Características incluidas en el DataFrame Original

Cómo se puede apreciar en la Figura 6.4, se cuenta con 5 características de tipo int64 tales como: Open time, Close time, Number of Trades, Ignore y la variable de salida Response. Asimismo, se cuenta con 8 características de tipo float64 tales como: Open, High, Low, Close, Volume, Quote asset volume, Taker buy base asset volume y Taker buy quote asset volume.

6.2.1 *Estadística Descriptiva*

A lo largo de este apartado se describen las distintas características que contiene el conjunto de datos con el objetivo de entender mejor los valores, rangos y distintos aspectos de cada característica. Algo importante a recalcar es que las características de Open time, Close time, Ignore y Response no están expresadas en las tablas que se presentan ya que son medidas de tiempo o por sus valores no muestran ninguna información significativa. La información es mostrada en 2 tablas, la primera se centra en los precios y la segunda en el volumen y distintas características relacionadas. Toda la información está dividida en horas, ya que la información inicialmente fue descargada bajo este periodo de tiempo.

	Open	High	Low	Close
count	7294.00	7294.00	7294.00	7294.00
mean	1727.33	1732.55	1721.96	1727.41
std	163.14	163.42	162.77	163.03
min	1192.34	1193.47	1190.22	1192.34
25%	1626.55	1630.99	1621.39	1626.66
50%	1742.06	1747.88	1736.45	1742.25
75%	1855.63	1860.20	1851.05	1855.63
max	2130.00	2141.30	2120.35	2129.23

Tabla 6.2: Tabla descriptiva de las Características relacionadas al precio.

Tal y como se puede apreciar en la Tabla 6.2 se cuenta con el mismo número de observaciones para los 4 diferentes precios. La media de precio de apertura por hora es de $1,727.33, para el precio alto es de $1.732.55. para el precio bajo es de $1.721.96 y para el precio de cierre es de $1.727.41. La desviación estándar para los precios por hora para las distintas características respectivamente son las siguientes: 163.14, 163.42 162.77 y 163.03. El valor mínimo por hora para el precio de apertura es de $1,192.34, para el precio alto es de $1,193.47, el precio bajo es de $1,190.22 y el precio de cierre es de $1,192.34.

En temas de cuartiles, para el primer cuartil se observa que por hora el precio de apertura es de $1,626.55, para el precio alto es de $1,630.99, para el precio bajo es de $1,621.93 y para el precio de cierres es de $1,626.66. El valor de la mediana por hora para el precio de apertura es de $1,742.06, para el precio alto es de $1,747.88, para el precio bajo es de $1,736.45 y para el precio de cierres es de $1,742.25. Para el tercer cuartil se observa que por hora los precios respectivamente son los siguientes: $1.855.63, $1,860.20, $1,851.05, $1,855.63.

Por último, el valor máximo por hora del precio de apertura es de $2,130.00, para el precio alto es de $2,141.30, para el precio bajo es de $2,120.35 y por último el precio de cierre es de $2,129.23.

	Volume	Quote asset volume	Number of trades	Taker buy base asset volume	Taker buy quote asset volume
count	7294.00	7.29e+03	7294.00	7294.00	7.29e+03
mean	4870.27	8.09e+06	6818.13	2448.08	4.07e+06
std	8140.48	1.30e+07	9562.16	4109.06	6.54e+06
min	0.00	0.00e+00	0.00	0.00	0.00
25%	603.52	1.05e+06	1381.25	307.62	5.28e+05
50%	1999.63	3.58e+06	3884.50	1000.20	1.78e+06
75%	5902.95	9.94e+06	8374.00	2945.97	5.02e+06
max	138580.47	2.11e+08	196974.00	74834.92	1.14e+08

Tabla 6.3: Tabla descriptiva de las Características relacionadas al volumen.

Ahora observando la Tabla 6.3 podemos determinar que de todas las características se tienen la misma cantidad de entradas. Tomando a la variable de volumen de transacciones, su media por hora es de 4,870.27 así como su desviación estándar es de 8,140.48. El valor mínimo por hora es de 0 mientras que el valor máximo por hora es de 138,580.47. Hablando sobre cuartiles, el valor del primer cuartil por hora es de 603.52, la mediana por hora es de 1,999.63 y el valor del tercer cuartil por hora es de 5,902.95.

Para la variable Quote asset volume o bien el volumen total del activo su media por hora es de 8.09e+06 y su desviación estándar es de 1.30e+07. El valor mínimo y máximo por hora es de 0 y 2.11e+08 respectivamente. Los valores de los cuartiles por hora son: primer cuartil 603.52, mediana 3.58e+06 y el tercer cuartil 9.94e+06.

Tomando la variable Number of trades, su media por hora es de 6,818.13 y su desviación estándar es de 9,562.16. Los valores mínimos y máximos por hora respectivamente son: 0 y 196,974. Los valores de los cuartiles son: primer cuartil 1,394.25, mediana 3,884.50 y tercer cuartil 8,374.

Los valores por hora que se muestran para la variable Taker buy base asset volume son los siguientes: media 2,448.08, desviación estándar 4,109.06, valor mínimo y máximo 0 y 74,834.92 respectivamente y los valores de los cuartiles son 307.62, 1000.20 y 2,945.97 respectivamente.

Por último, la varaiable Taker buy quote asset volume tiene una media por hora de 4.07e+06, una desviación estándar de 6.54e+06, un valor mínimo por hora de 0 mientras que el valor máximo por hora es de 1.14e+08. Los valores de los cuartiles son: primer cuartil 5.28e+05, mediana 1.78e+06 y tercer cuartil de 5.02e+06.

Ya que se han presentado estadísticas como media, desviación, cuartiles, valores máximos y valores mínimos queda por describir la asimetría y alargamiento de las variables.

Característica	Valor
Open	-0.624179
High	-0.626043
Low	-0.618235
Close	-0.622225
Volume	4.708800
Quote asset volume	4.493413
Number of trades	4.832944
Taker buy base asset volume	4.760273
Taker buy quote asset volume	4.527951
Ignore	0.000000
Response	-0.009873

Tabla 6.4: Simetría de las Características del DataFrame

Tal y como se puede observar en la Tabla 6.4 las variables Open, High, Low, Close tienen valores menores que 0. Variables como Ignore y Response tienen valores iguales a 0 o muy cercanos (como es el caso de Response). Por otro lado, Volume, Quote asset Volume, Number of trades, Taker buy base asset volume y Taker buy quote asset volume tienen valores mayores a 0.

Esto quiere decir que las variables relacionadas a los precios (las que son menores que 0 son asímetricas con una cola alargada hacia la izquierda mientras que las variables relacionadas al volumen (las que son mayores que 0) también son asimétricas pero con la cola alargada hacia la derecha.

Característica	Valor
Open	0.725154
High	0.754403
Low	0.683988
Close	0.721330
Volume	36.078795
Quote asset volume	32.755534
Number of trades	44.204848
Taker buy base asset volume	37.809828
Taker buy quote asset volume	34.043795
Ignore	0.000000
Response	-2.000451

Tabla 6.5: Curtosis de las Características del DataFrame

Ahora observando la Tabla 6.5 podemos revisar los valores de apuntamiento o curtosis de las variables del conjunto de datos. Variables como Open, High, Low y Close presentan valores entre 0 y 1, son variables platicúrticas. Variables como Volume, Quote asset volume, Number of trades, Taker buy base asset volume y Taker buy quote asset volume tienen valores mayores a 30, son variables leptocúrticas. Response tiene un valor menor a 0 e Ignore tiene un valor igual

a 0, por lo que son variables platicúrticas.

6.2.2 *Análisis exploratorio*

En este apartado se muestran distintas visualizaciones con el objetivo de entender mejor la relación entre las variables, para resolver dudas que pudieran responderse de manera visual y para complementar la descripción de las variables.

Figura 6.5: Visualización del conjunto de datos en formato de Velas

Inicialmente, tal y como se muestra en la Figura 6.5, se puede observar todo el conjunto de datos en formato de velas, que es la manera común de visualizarlo. A grandes rasgos se puede determinar que existen incrementos y decrementos en el activo que se analiza en este proyecto. Otro aspecto que se puede determinar es que desde comienzo del año 2023 hasta el fin de la fecha disponible se visualiza un incremento general en el precio que podemos observar de forma cuantitativa en la Tabla 6.2 donde se muestra el valor mínimo y valor máximo del precio de cierre.

6.2.3 *Visualización de Distribución de las Variables*

Como parte del complemento de la descripción de las variables, a continuación se muestran histogramas para representar las variables.

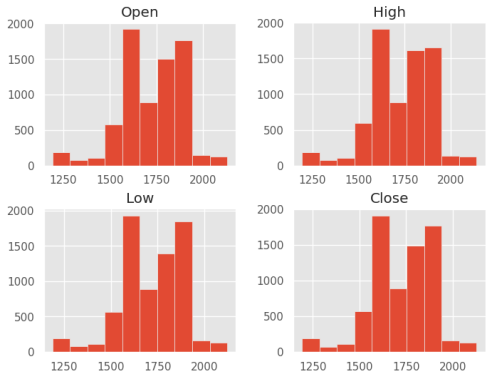

Figura 6.6: Histogramas de las Variables Open, High, Low y Close del Conjunto de Datos

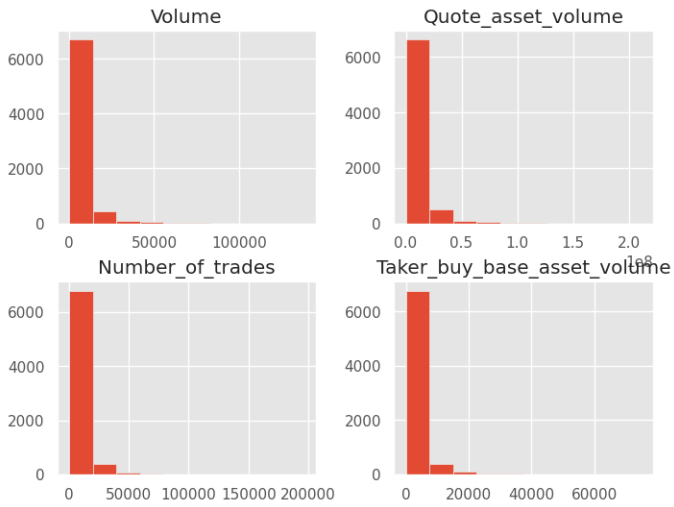

Figura 6.7: Histogramas de las Variables Volume, Quote asset volume, Number of trades y Taker buy base asset volume

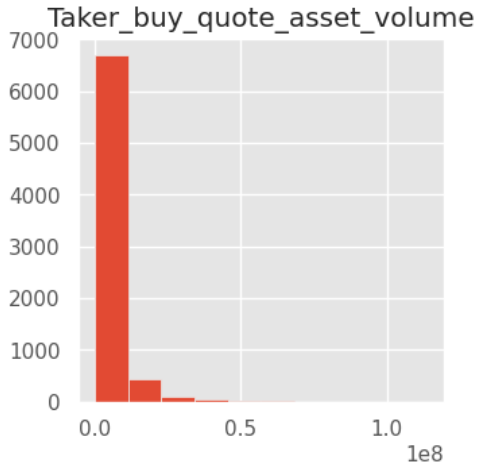

Figura 6.8: Histogramas de la Variable Taker buy quote asset volume

Observando la Figura 6.6, Figura 6.7 y Figura 6.7 podemos relacionar estadísticas que previamente se han revisado, tales como las que aparecen en la Tabla 6.4, en ella podemos conocer de forma cuantitativa el valor de la simetría de las distintas variables. En las variables con los valores mayores a 1 se puede apreciar con facilidad la asimetría. Mientras que las variables que tienen valores cercanos a 0 es más difícil apreciar la simetría de forma visual. Asimismo podemos determinar de forma visual lo que se expresó previamente en la Tabla 6.5, los valores mayores a 30 se aprecian con facilidad el apuntamiento en la distribución del conjunto de los datos.

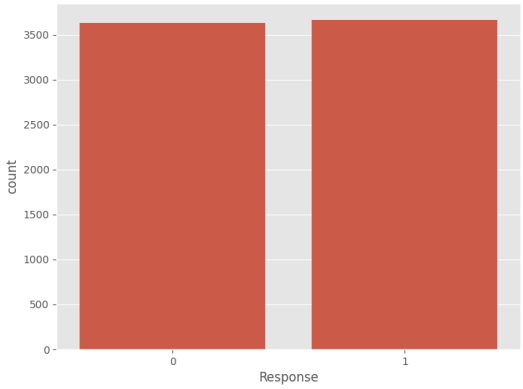

Figura 6.9: Distribución de la Variable Objetivo

Utilizando la Figura 6.9 como punto de referencia, pareciera ser que están al mismo nivel. Es por ello que se complementa de forma cuantitativa tal y como aparece en la tabla de abajo.

Respuesta	Valor
0	3,665
1	3,629

Tabla 6.6: Conteo de Resultados

Con la ayuda de la Figura 6.9 en conjunto con la Tabla 6.6 podemos determinar que en el periodo de tiempo analizado, el decremento de precio ocurre 36 veces más que el incremento.

6.2.4 *Análisis de relación entre Variables de Tiempo y el Incremento de Precio*

Ahora bien, en la siguiente parte del Análisis Exploratorio se describe la relación del incremento o decremento de precio por parte del activo con distintas variables de tiempo tales como: día de la semana, hora del día y mes del año.

Comenzando por hora del día, la figura que se muestra a continuación es un conteo de los incrementos o decrementos de precio tomando como referencia el tiempo previamente descrito.

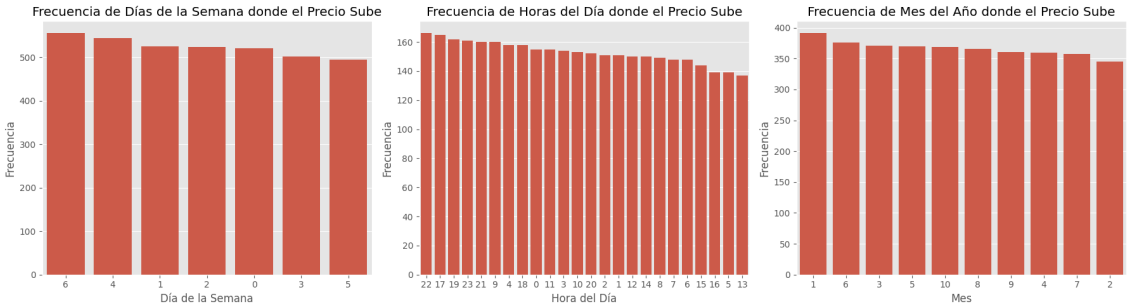

Figura 6.10: Frecuencia de Incremento de Precio por Hora del Día, Día de la Semana y Mes del Año

En la Figura 6.10 se aprecia que las 3 horas con mayor frecuencia de incremento de precio son 22, 17 y 19 mientras que las 3 horas con menor frecuencia de incremento de precio son 16, 5 y 13. A simple vista, comparando una hora antes contra una hora después no se nota un gran cambio. No es hasta que comparamos los de mayor frecuencia contra los de menor frecuencia donde se encuentra una diferencia notable. Es por ello que a continuación se agrega una tabla con el conteo de incrementos para poder comparar de manera cuantitativa.

Hora	Valor	Hora	Valor
0	155	1	151
2	151	3	154
4	158	5	139
6	148	7	148
8	149	9	160
10	153	11	155
12	150	13	137
14	150	15	144
16	139	17	165
18	158	19	162
20	152	21	160
22	166	23	161

Tabla 6.7: Frecuencia de Incrementos por Hora

Ahora bien, con base en la Tabla 6.7 podemos determinar que la hora con mayor frecuencia, las 22 horas, es mayor en 29 incidencias que las 13 horas, la hora con menor incidencia. Si comparamos las tres horas con mayor incidencias (22, 17 y 19) podemos observar que la diferencia

es de 1 o 4 incidencias. Si observamos a detalle, las primeras cinco horas con mayor incidencia son pasando el medio día (22, 17, 19, 23 y 21).

Una vez analizada la hora del día, se analiza el día de la semana tal y como se muestra en la figura de abajo.

Se aprecia en la Figura 6.10 qué los días de mayor incidencia son 6, 4 y 1. A continuación se incluye una tabla que muestra de manera cuantitativa la misma frecuencia.

Día	Valor
0	521
1	525
2	524
3	502
4	544
5	494
6	555

Tabla 6.8: Frecuencia de Incrementos por Día

Bajo los valores de la Figura 6.10 observamos que el día 6 tiene una frecuencia de 555 mientras que el día 4 de 544, 11 incidencias de diferencia. Existe una diferencia de 61 incidencias entre el día con mayor frecuencia y el día de menor frecuencia.

Ahora bien, se procede a analizar la frecuencia de incremento de precio a lo largo de los meses contenidos en el conjunto de datos.

En la Figura 6.10 se aprecia que los meses de mayor incidencia son 1, 6 y 3. A continuación se incluye una tabla que muestra de manera cuantitativa la misma frecuencia.

Mes	Valor
1	391
2	345
3	371
4	359
5	370
6	376
7	357
8	366
9	361
10	369

Tabla 6.9: Frecuencia de Incrementos por Mes

Tomando en cuenta los valores de la Tabla 6.9 observamos que el mes 1 tiene una frecuencia de 391 mientras que el mes 6 (segundo mes con mayor frecuencia] de 376, 15 incidencias de diferencia. Existe una diferencia de 46 incidencias entre el mes con mayor frecuencia y el mes de menor frecuencia.

6.2.5 *Detección de Atípicos*

En este apartado se analizó, mediante el uso de *boxplots* o diagramas de caja, la existencia de datos atípicos. Aunado a esto, se agregan tablas para complementar de una manera cuantitativa.

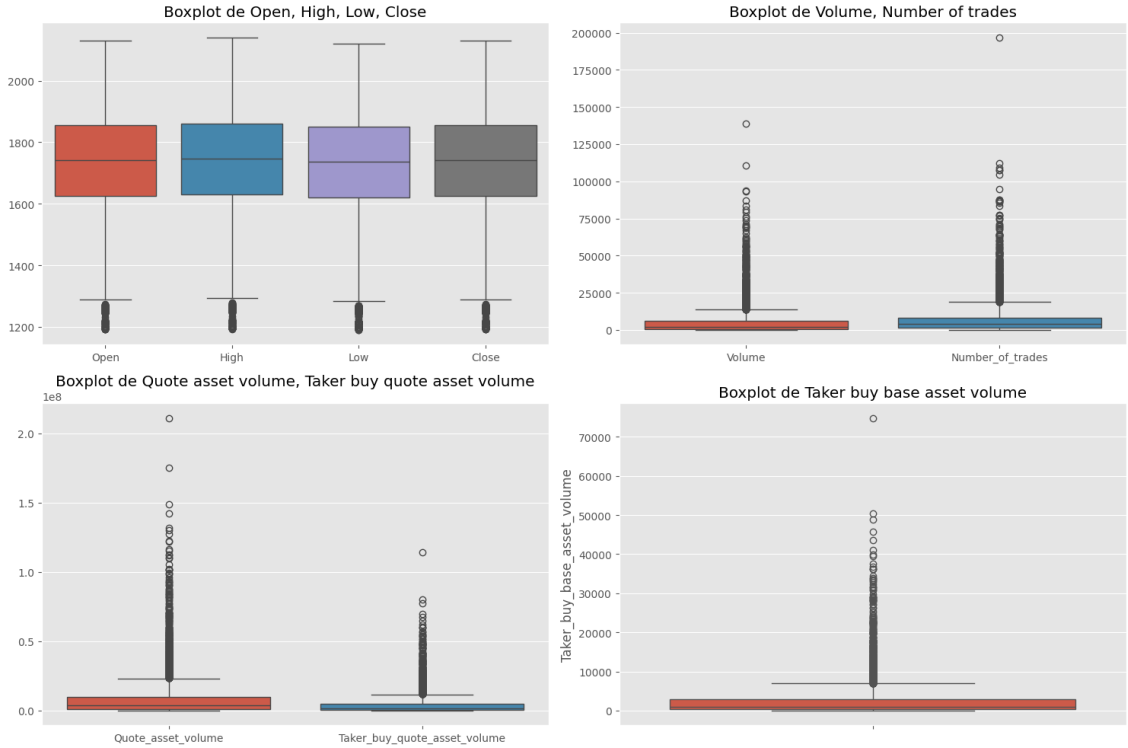

Figura 6.11: Boxplot de Precios, Volumen y Número de Transacciones, Volumen del Activo y Volumen de Compra de Activo, Volumen de la base

Como se observa en la Figura 6.11 hay 4 *boxplots*, 1 para cada variable de precio, podemos determinar que a simple vista existen similitudes entre los distintos precios. Se aprecia que los valores de los cuartiles entre los 4 precios rondan valores similares así como también es el caso del rango intercuartílico.

Retomando la Tabla 6.2, la mediana de los 4 precios va de $1,736.45 hasta $1747.88, es por ello que en los gráficos se observa similitud entre ellos. Asimismo, los valores del cuartil 1 van de $1,621.39 hasta $1630.99 y los valores del cuartil 3 va de $1,851.05 hasta $1,860.20. Es por ello que a simple vista se observan similares los gráficos.

Ahora bien, observamos que en la parte inferior del gráfico se encuentra una gran cantidad de datos atípicos. Es por ello que se agrega la Tabla 6.10 que contiene los límites inferiores y superiores de cada variable. Los datos que son menores al límite inferior son datos atípicos.

Variable	Límite Inferior	Límite Superior
Open	1.28e+03	2.20e+03
High	1.29e+03	2.20e+03
Low	1.28e+03	2.20e+03
Close	1.28e+03	2.20e+03

Tabla 6.10: Tabla de Límites Inferior y Superior de las Variables de Precios

Variable	Límite Inferior	Límite Superior
Volume	-7.35e+03	1.39e+04
Number of Trades	-9.11e+03	1.89e+04

Tabla 6.11: Tabla de Límites Inferior y Superior de las Variables relacionadas a Volumen

Por otra parte, tal y como se observa en la Figura 6.11, sucede lo contrario a la figura anterior. En esta figura, las variables de Volumen y Número de Transacciones, contienen datos atípicos en la parte superior, o bien, usando la Tabla 6.11, los datos atípicos son aquellos superiores a 1.39e+04 para el caso de Volumen y 1.89e+04 para el caso de Número de Transacciones.

Variable	Límite Inferior	Límite Superior
Quote asset volume	-1.23+07	2.33e+07
Taker buy quote asset volume	-6.21e+06	1.18e+07

Tabla 6.12: Tabla de Límites Inferior y Superior de las Variables relacionadas a Volumen del Activo

Para las variables de Quote asset volume y Taker buy quote asset volume, la presencia de datos atípicos esta ubicado en la parte superior de la Figura 6.11. En este caso, los datos atípicos son mayores a 2.33e+07 y 1.18e+07 respectivamente, tal y como lo muestra la Tabla 6.12.

Por último, con base en la Figura 6.11, observamos un comportamiento similar a las anteriores variables ya que se aprecia que los datos atípicos están en dirección hacia el límite superior. Utilizando la Tabla 6.13, los datos atípicos son mayores a 6.90e+03.

Variable	Límite Inferior	Límite Superior
`Taker buy base asset volume`	-3.65e+03	6.90e+03

Tabla 6.13: Tabla de Límites Inferior y Superior de las Variables relacionadas a Volumen de la base

6.2.6 *Ajuste de Distribución de Probabilidad Emírica*

Con base en la información de la Tabla 6.2 y Tabla 6.3, podemos observar que en la primera tabla la media y la mediana tienen valores similares. Es por ello que se da por hecho que tienen una distribución normal. Sin embargo, la Tabla 6.3 no muestra la misma similitud entre sus valores como la tabla de precios. Es por ello que se ajustan las distribuciones de probabilidad empírica.

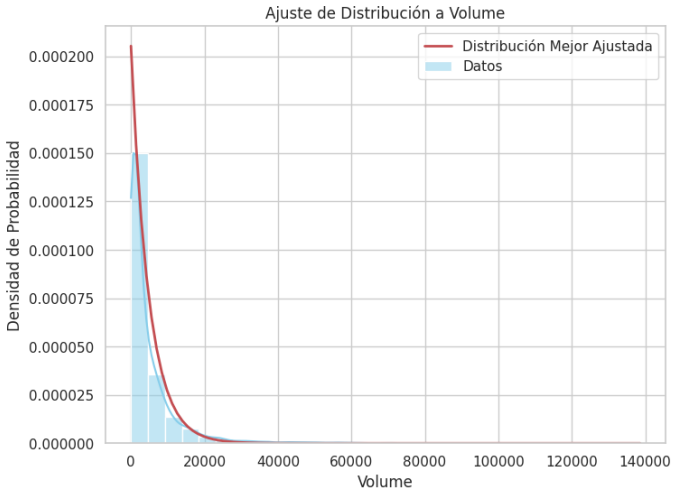

Figura 6.12: Distribución de variable Volume

Distribución	SSE	AIC	BIC
`gamma`	2.530898e+34	512932.188959	512952.873381
`expon`	6.562986e+11	133594.152064	133607.941679
`lognorm`	6.562986e+11	133596.152067	133616.836489
`beta`	6.562986e+11	133613.939887	133641.519117
`norm`	6.562986e+11	133594.152075	133607.941690

Tabla 6.14: Tabla de Resultados de Ajuste de Probabilidad Emírica de Volume

Como se aprecia en la Figura 6.12, se realizó el ajuste de distribución tomando en cuenta distintas distribuciones como: Gamme, Exponencial, Lognormal, Beta y Normal. Para elegir la mejor distribución se usa SSE, AIC y BIC. Para el caso de volumen se observa que la mejor distribución con base en el resultado de SSE, AIC y BIC es la exponencial.

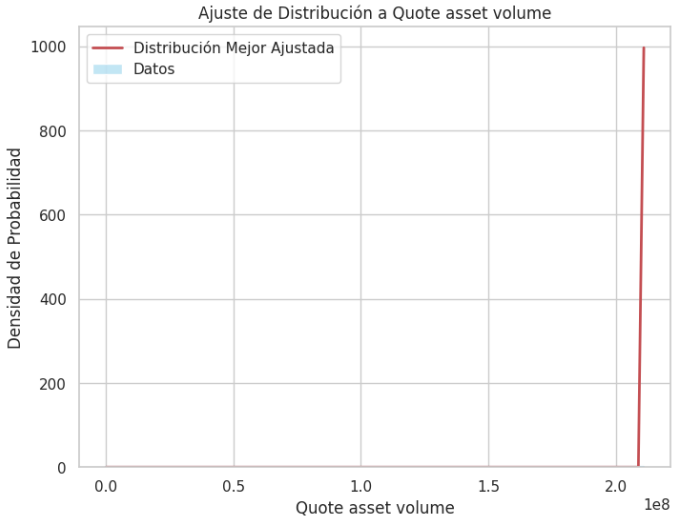

Figura 6.13: Distribución de variable Quote asset volume

Distribución	SSE	AIC	BIC
`gamma`	1.705955e+18	241334.183714	241354.868136
`expon`	1.705955e+18	241332.185513	241345.975128
`lognorm`	1.705955e+18	241332.185513	241345.975128
`beta`	1.705955e+18	241332.185513	241345.975128
`norm`	1.705955e+18	241332.185513	241345.975128

Tabla 6.15: Tabla de Resultados de Ajuste de Probabilidad Empírica de Quote asset volume

Aún y cuando visualmente no se puede apreciar el ajuste de distribución de la Figura 6.13 se elige con base en las métricas. De acuerdo a los valores de la Tabla 6.15 se elige la distribución exponencial ya que los valores de SSE, AIC y BIC son los menores a comparación de los demás.

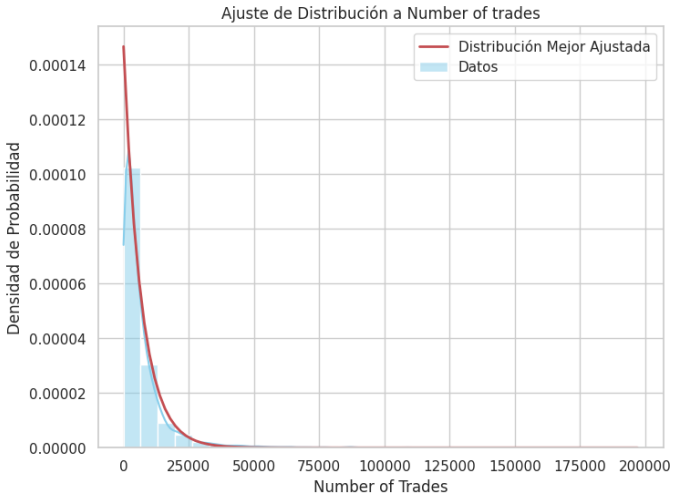

Figura 6.14: Distribución de variable Number of trades

Distribución	SSE	AIC	BIC
`gamma`	2.533528e+20	277808.982999	277829.667421
`expon`	1.005910e+12	136708.926003	136722.715618
`lognorm`	1.005910e+12	136708.926003	136731.610425
`beta`	1.005910e+12	136712.926004	136740.505233
`norm`	1.005910e+12	136708.926010	136722.715624

Tabla 6.16: Tabla de Resultados de Ajuste de Probabilidad Emírica de Number of Trades

De acuerdo con lo que se observa en la Figura 6.14 y la Tabla 6.16 se determina que la mejor distribución es la exponencial ya que los valores de SSE, AIC y BIC son los menores.

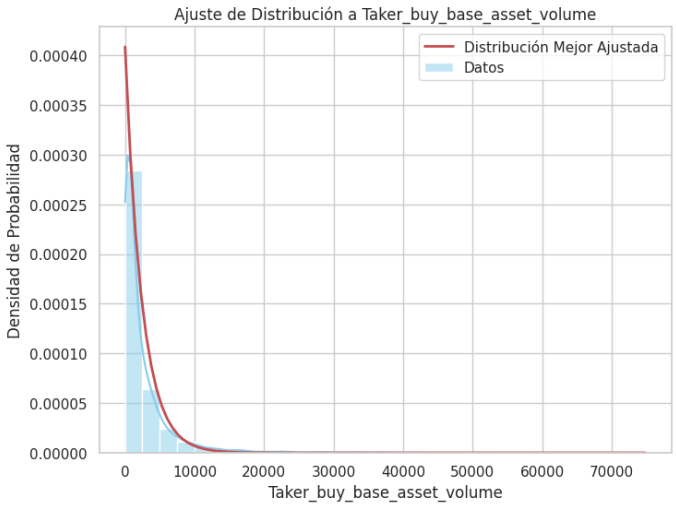

Figura 6.15: Distribución de variable Taker buy base asset volume

Distribución	SSE	AIC	BIC
gamma	4.622806e+32	483736.183462	483756.867885
expon	1.668511e+11	123604.917283	123618.706898
lognorm	1.668511e+11	123606.917294	123627.601716
beta	1.668511e+11	123608.917295	123636.496524
norm	1.668511e+11	123604.917325	123618.706940

Tabla 6.17: Tabla de Resultados de Ajuste de Probabilidad Emírica de Taker buy base asset volume

Se determina que tomando en cuenta la Figura 6.15 y la Tabla 6.17, para la variable Taker buy base asset volume la mejor distribución es la exponencial.

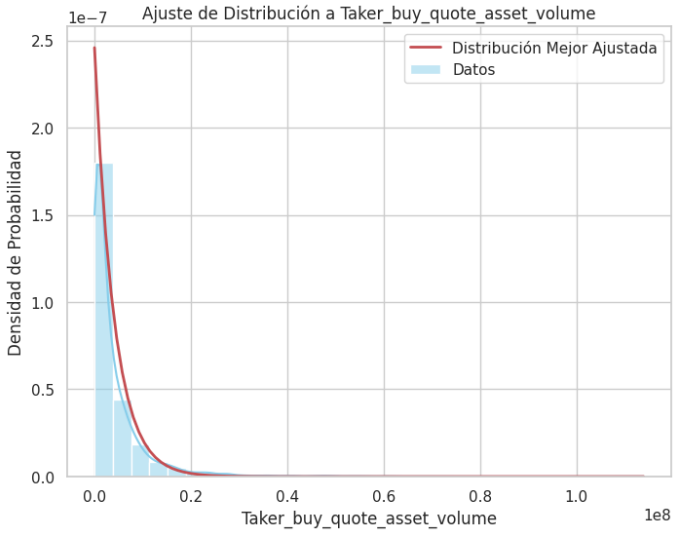

Figura 6.16: Distribución de variable Taker buy quote asset volume

Distribución	SSE	AIC	BIC
gamma	4.328893e+17	231331.205356	231351.889778
expon	4.328893e+17	231331.205356	231342.994971
lognorm	4.328893e+17	231331.205356	231351.889778
beta	4.328893e+17	231331.205356	231360.784585
norm	4.328893e+17	231331.205356	231342.994971

Tabla 6.18: Tabla de Resultados de Ajuste de Probabilidad Emírica de Taker buy quote asset volume

Por último, analizando la Figura 6.16 y la Tabla 6.18 se determina que la mejor distribución para la variable Taker buy quote asset volume es la distribución exponencial. Por un lado, las variables relacionadas a precio que por su estadística descriptiva se determina que su distribución se asemeja a la distribución normal y por otro lado, las variables relacionadas a volumen, una vez ajustadas, se determina que su distribución se asemeja a la distribución exponencial.

Tomando en cuenta la información antes presentada de valores atípicos de las variables, así como el ajuste de distribución de probabilidad empírica que tiene como resultado distribuciones exponenciales se determina que, partiendo de la volatilidad inherente en las variables, es menos común que los precios tengan un valor menor a $1,621.39. Por parte del Volumen y el Número de transacciones, no es común que superen los 13,900 y 18,900 respectivamente. Con variables como Quote asset volume y Taker buy quote asset volume los valores comúnmente no superan 23,300,000 y 11,800,000 respectivamente. Por último, no es común que la variable Taker buy base asset volume tenga valores superiores a 6,900. Esto analizando las gráficas previamente presentadas.

Ahora bien, con lo anterior y analizando que las variables tienen una distribución de probabilidad empírica exponencial, existe la posiblidad de tener valores muy altos, sin embargo la mayor cantidad de datos está ubicado en valores menores. Con esto se reafirma que los datos del activo utilizado son volatiles y deben ser considerados para la generación de características.

6.2.7 *División de Sub-conjuntos*

En procesos de pronóstico de series temporales financieras, muy frecuentemente se trabaja con datos que representan una única muestra de un proceso estocástico, cuya representación exacta es imposible de conocer por completo debido a que el proceso generador de muestras es de naturaleza estocástica secuencial, hace que sea imposible generar muestras de variaciones de ese proceso en el pasado. Esta es una razón principal del por qué trabajar con series de tiempo financieras para modelado predictivo sea un reto fundamentalmente diferente de otros contextos y tipos de datos.

Para abordar esta cuestión, se ha documentado el emplear técnicas que implican generadores de subconjuntos de datos acotados en el tiempo. Estas técnicas dividen el conjunto de datos principal en sub muestras, cada una de las cuales se construye con la finalidad de capturar diferentes distribuciones de probabilidad. Como se muestra en este trabajo [2], la división del conjunto de datos en sub-conjuntos llamados "Folds", implementado tanto para variables explicativas como objetivo, muestra una mejora en la representación de las muestras de la desconocida distribución del proceso generador de datos, facilitando la generalización fuera de la muestra e incluso fuera de la distribución en determinadas condiciones.

[2] J. D. Muñoz-Elguezábal, J. F. Sánchez-Torres, "T-fold sequential-validation technique for out-of-distribution generalization with financial time series data. international conference on econometrics and statistics," 2021

6.2.8 *Ingeniería de Características*

Para el desarrollo del modelado predictivo es necesario crear características que ayuden al análisis ya que las variables antes explicadas no son suficientes. La creación de características se divide en 2 partes, la primera son características lineales y la segunda son características autorregresivas.

Características Lineales

Las características lineales que se crean son las diferencias de precios de las distintas variables como Open, High, Low y Close. Esta diferencia sucede en el mismo tiempo, es decir se restan los precios de la misma hora. El objetivo es representar los cambios de precios de forma lineal y de esta forma poder proporcionar información de tendencias simples a los modelos. Otro punto importante de estas características es partiendo del análisis exploratorio previamente expuesto, ya que debido a la alta volatilidad es necesario poder entender las tendencias de los precios.

Las características lineales son las siguientes:

`CO`	Close - Open
`HL`	High - Low
`OL`	Open - Low
`HO`	High - Open
`COV`	Close - Open / Volume
`HLV`	High - Low / Volume

Tabla 6.19: Tabla demostrativa de las Características lineales.

A continuación se presenta un análisis de las características lineales:

Tal y como se aprecia en la Figura 6.17, al restar la variable Close y Open los datos se agrupan en una distribución similar a la normal. Al ser una característica lineal no existe inconveniente en que sean los mismos datos.

Ahora bien, visualizando la Figura 6.17 se observa una distribución similar a la exponencial, tal y como las variables previamente analizadas. Se observa una mayor concentración de datos al inicio de la gráfica y lo podemos conectar con la estadística descriptiva ya que los valores High y Low no están tan alejados y eso se refleja en esta figura.

En la Figura 6.17 se observa un comportamiento similar a lo que se observa en la **??**. A simple vista su distribución es cercana a la exponencial y tomando en cuenta la estadística descriptiva se reafirma lo mismo que en la característica anterior.

Tal y como se observa en la Figura 6.17 hay un comportamiento similar que en la **??**. Al final tenemos 2 características con un comportamiento similar y otras 2 que visualmente se observan parecidas.

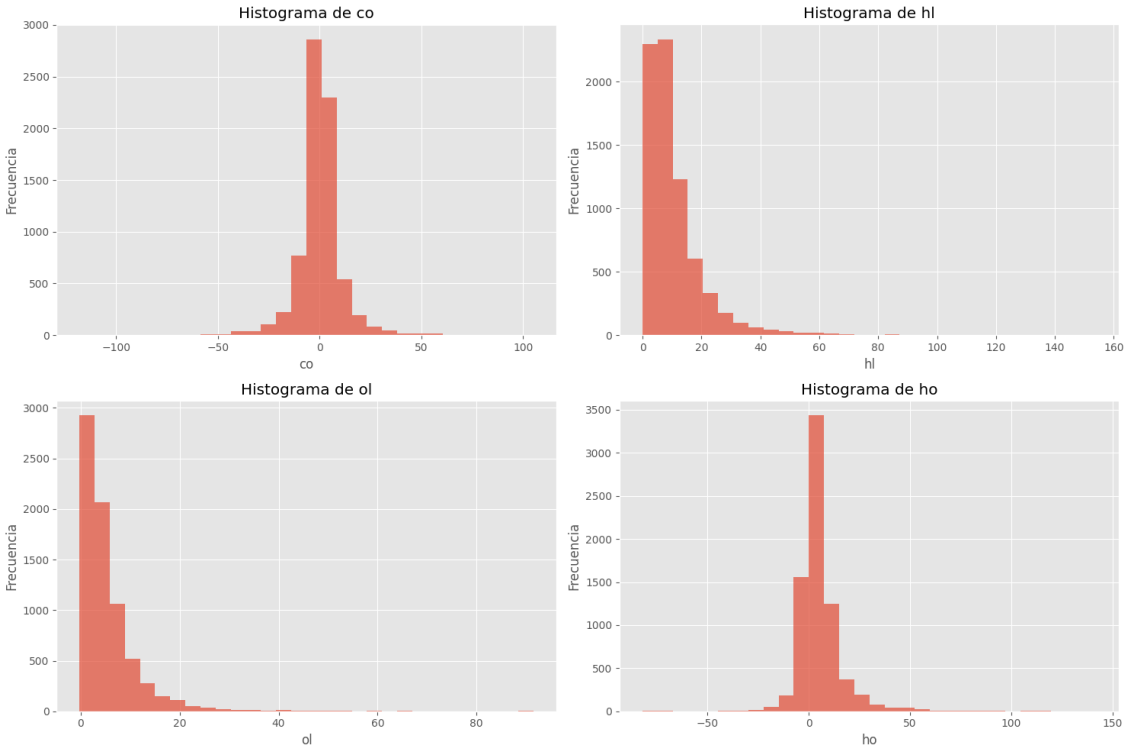

Figura 6.17: Distribución de Características Lineales

Características Autorregresivas

Como ya se presentó previamente, las características lineales tienen como objetivo capturar la tendencia de los distintos precios en la misma instancia. Sin embargo, es necesario capturar tendencias usando información pasada, es por ello que se crean las características autorregresivas. Se crean promedios móviles, desviaciones y rezagos utilizando las características lineales. Se proponen 2 grupos de estas características, el primero considera 2 instancias y el segundo considera 3. Esto con el fin de capturar tendencias pasadas.
Las características autorregresivas son las siguientes:

- **ma ol n: promedio móvil de la característica OL usando n instancias**
- **ma ho n: promedio móvil de la característica HO usando n instancias**
- **ma hl n: promedio móvil de la característica HL usando n instancias**
- **ma hlv n: promedio móvil de la característica HLV usando n instancias**
- **ma cov n: promedio móvil de la característica COV usando n instancias**
- **sd ol n: desviación de la característica OL usando n instancias**
- **sd ho n: desviación de la característica HO usando n instancias**
- **sd hl n: desviación de la característica HL usando n instancias**
- **sd hlv n: desviación de la característica HLV usando n instancias**

- **sd cov n: desviación de la característica COV usando n instancias**
- **lag vol n: rezago de la característica Volume en n instancias**
- **sum vol n: suma móvil de Volume en n instancias**
- **mean vol n: promedio móvil de Volume en n instancias**

Con estas características se busca capturar las tendencias entre el mismo tiempo, considerando tiempos anteriores y de esta forma poder predecir el cambio de precio en el tiempo futuro.

6.2.9 *Pre-procesamiento general de datos*

En este apartado, se deben escalar los datos. Para el caso de este proyecto se utilizó *MinMaxScaler* de sklearn. Esto es importante por distintas razones:

- Existen valores átipicos en algunas variables como se explicó en la subsección *Datos Atípicos*, al escalar los valores se busca que no afecten al rendimiento del modelo
- Mejorar la estabilidad del modelo al evitar las distintas escalas de los valores
- Igualar la importancia relativa de las variables

6.2.10 *Tratamiento de datos faltantes*

Al crear las características lineales, se retraza 1 instancia para evitar utilizar la misma información y ser profeta del pasado, esto genera datos faltantes. Se elimina la entrada completa, reduciendo en 1 la dimensión del conjunto de datos.

Al crear las características autorregresivas, sucede algo similar que con las características lineales. En este caso los datos faltantes son reemplazados con 0 ya que si se eliminan las entradas, el dataframe se reduce en gran cantidad.

6.3 *Metodología*

La metodología utilizada es la siguiente:

1. Obtención de datos intradía con frecuencia de 1 hora de la criptomoneda Etherium del portal Binance Vision.
2. Análisis Descriptivo y Exploratorio de los datos.
3. Ingeniería de Características.
4. Preprocesamiento del conjunto de datos.
5. Escalamiento del conjunto de datos.
6. Uso de modelos de machine learning.
7. Evaluación de modelos.

6.3.1 *Descripción de los Modelos*

En este apartado se detalla la descripción de los modelos utilizados de *Machine Learning* para el desarrollo del sistema de modelado predictivo en el mercado de criptomonedas (tal y como se expresa en el objetivo general).

Perceptrón Multicapa Es un tipo de Red Neuronal que tiene múltiples capas de neuronas (capa de entrada, una o más capas ocultas y una capa de salida).

Regresión Logística Modelo que se utiliza para predecir la probabilidad de pertenencia a una clase.

La selección de los modelos partió de la necesidad de hacer predicciones tomando en cuenta el conjunto de datos a utilizar en la que la respuesta debe ser binaria (cambio de signo de la función objetivo). Asimismo, tomando en cuenta la perspectiva que toma David Spiegelhalter [3] que se ha abordado anteriormente, se quiere comparar bajo este problema que tipo de modelo puede resolverlo con mayor certeza, ya sea un modelo basado en regresión o lógica basada en reglas.

[3] D. Spiegelhalter, *The Art of Statistics How to Learn from Data*. Basic Books, 1st ed., March 2021

6.3.2 *Descripción de las Métricas*

En esta sección se presentan las métricas utilizadas para evaluar la eficacia de los modelos utilizados.

6.3.3 *Matriz de Confusión*

Se agrega la matriz de confusión como ayuda para entender las fórmulas.

		Actual	
		Positivo	**Negativo**
Predicciónl	**Positivo**	Verdadero Positivo VP	Falso Negativo FN
	Negativo	Falso Positivo FP	Verdadero Negativo VN

6.3.4 *Exactitud (Accuracy)*

$$Exactitud = \frac{\text{VP + VN}}{\text{VP+VN+FP+FN}} \tag{6.1}$$

6.3.5 *Precisión (Precision)*

$$Precisin = \frac{\text{VP}}{\text{VP+FP}} \tag{6.2}$$

6.3.6 *Sensibilidad o Tasa de Verdaderos Positivos (Recall)*

$$Sensibilidad = \frac{\text{VP}}{\text{VP+FN}} \tag{6.3}$$

6.3.7 *Valor F1 (F1-Score)*

$$ValorF1 = 2\frac{\text{Precisión x Sensibilidad}}{\text{Presición + Sensibilidad}} \tag{6.4}$$

6.3.8 *Área Bajo la Curva ROC (AUC-ROC)*

Es necesario construir la Curva Roc y se compone por 2 valores: Tasa de Verdaderos Positivos o Recall y Tasa de Falsos Positivos. La formula de Tasa de Falsos Positivos es la siguiente:

$$TasadeFalsosPositivos = \frac{\text{FP}}{\text{FP + VN}} \tag{6.5}$$

Para calcular el área bajo la curva ROC, se grafica la curva y después se suma el área total debajo de la curva.

Estas métricas son relevantes ya que para evaluar y comparar los rendimientos de los modelos utilizados es importante maximizar la exactitud de las predicciones de los cambios de signo de la función objetivo de este proyecto. Asimismo, se busca encontrar un equilibrio entre la precisión y la sensibilidad para evitar sesgos hacia alguna clase.

6.3.9 *Hipótesis*

La variable *volumen* es importante para el cambio de signo de la variable objetivo. La relación es lineal y positiva con respecto al cambio de signo de la variable objetivo.

6.3.10 *Diseño de experimento 1*

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales
Entradas	7,294, 17 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Tamaño del Batch, Optimizador y Número de Épocas
Preprocesamiento	Escalamiento

Tabla 6.20: Tabla de Diseño de Modelo Perceptrón Multicapa Experimento 1

En la Tabla 6.20 se muestran las categorías y los valores del experimento 1 usando Perceptrón Multicapa.

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales
Entradas	7,293, 17 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Solvers, Penalty, C Values
Preprocesamiento	Escalamiento

Tabla 6.21: Tabla de Diseño de Modelo de Regresión Logística Experimento 1

En la Tabla 6.21 se muestran las categorías y los valores del experimento 1 usando Regresión Logística.

6.3.11 *Diseño de experimento 2*

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Tamaño de Capa Oculta, Función de Activación, Solver, Alpha, Tamaño del Batch, Tasa de Entrenamiento
Preprocesamiento	Escalamiento

Tabla 6.22: Tabla de Diseño de Modelo Perceptrón Multicapa Experimento 2

En la Tabla 6.22 se muestran las categorías y los valores del experimento 2 usando Perceptrón Multicapa.

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Solvers, Penalty, C Values
Preprocesamiento	Escalamiento

Tabla 6.23: Tabla de Diseño de Modelo de Regresión Logística Experimento 2

En la Tabla 6.23 se muestran las categorías y los valores del experimento 2 usando Regresión Logística.

6.3.12 *Realización de experimento 1*

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales
Entradas	7,293, 17 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Tamaño del Batch-50; Número de Épocas-50; Optimizador-adam
Preprocesamiento	Escalamiento con MinMaxScaler

Tabla 6.24: Tabla de Modelo Perceptrón Multicapa Experimento 1

En la Tabla 6.24 se muestran las categorías y los valores reales del experimento 1 usando Perceptrón Multicapa.

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Solvers-liblinear; Penalty-l2; C Values-10
Preprocesamiento	Escalamiento con MinMaxScaler

Tabla 6.25: Tabla de Modelo de Regresión Logística Experimento 1

En la Tabla 6.25 se muestran las categorías y los valores reales del experimento 1 usando Regresión Logística.

6.3.13 *Realización de experimento 2*

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Tamaño de Capa Oculta-500,500; Función de Activación-tanh; Solver-adam
Hiperparámetros	Alpha-0.0001; Tamaño del Batch-100, Tasa de Entrenamiento-constante; early stopping-Verdadero
Preprocesamiento	Escalamiento con MinMaxScaler

Tabla 6.26: Tabla de Modelo Perceptrón Multicapa Experimento 2

En la Tabla 6.26 se muestran las categorías y los valores reales del experimento 2 usando Perceptrón Multicapa.

Categoría	Valor
Librerías	Pandas, Numpy, Datetime, seaborn, matplotlib, sklearn
Datos	7,294 entradas, 13 variables
Ingeniería de Características	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Hiperparámetros	Solvers-liblinear; Penalty-l1; C Values-1
Preprocesamiento	Escalamiento con MinMaxScaler

Tabla 6.27: Tabla de Modelo de Regresión Logística Experimento 2

En la Tabla 6.27 se muestran las categorías y los valores reales del experimento 2 usando Regresión Logística.

7 Resultados

En este capítulo se presentan los resultados obtenidos. Se divide en dos secciones: Resultados obtenidos y Análisis y Discusión de resultados.

7.1 Resultados obtenidos

Resultados Primer y Segundo Ciclo				
Métrica / Modelo	Red Neuronal	Regresión Logística	Perceptrón Multicapa	Regresión Logística
Accuracy	0.50	0.52	0.85	0.79
Precision	0.49	0.51	0.88	0.81
Recall	0.46	0.30	0.82	0.75
AUC-ROC	0.51	0.53	0.85	0.88

Tabla 7.1: Tabla de Resultados obtenidos en el Primer y Segundo Ciclo

Los valores presentados en las primeras 3 columnas de la Tabla 7.1 son los resultados obtenidos separados por métricas de cada modelo. Los resultados obtenidos con Regresión Logística son superiores a los de la Red Neuronal por 0.02 en casi todas las métricas. Sin embargo, los valores obtenidos por parte de los 2 modelos son deficientes ya que se obtuvó una Acertividad alrededor del 50 %.

Los valores presentados en las últimas 2 columnas de la Tabla 7.1 son los resultados obtenidos de cada modelo en el segundo ciclo. Los resultados obtenidos del Perceptrón Multicapa son superiores a los de la Regresión Logística en 6 unidades, comparando métrica por métrica. A comparación con el primer ciclo, la acertividad aumentó en más de 30 puntos.

7.2 Análisis y Discusión de resultados

En el siguiente apartado se expresan las diferencias entre cada ciclo divididos en distintas subsecciones.

7.2.1 Composición Primer Ciclo

En el primer ciclo el conjunto de datos utilizado contenía 17 características distintas. Estas características son: Open, High, Low, Close, Volume, Quote asset volume, Number of trades, Taker buy base asset volume, Taker buy quote asset volume, Response, day of the week, hour of the day, month, daily range, buy sell ratio, average volume, relative volume.

Los hiperparámetros utilizados en el modelo de Red Neuronal son: batch size, epochs, y optimizer. El número de neuronas en la capa inicial son el mismo número de características y contiene 1 capa oculta. Estas decisiones no están fundamentadas, fueron meramente arbitrarias.

Hiperparámetro	Valor
batch size	50
epochs	50
optimizer	Adam

Tabla 7.2: Tabla de Hiperparámetros Modelo Red Neuronal Primer Ciclo

Los hiperparámetros utilizados en el modelo de Red Neuronal son los que muestra la Tabla 7.2

Los hiperparámetros utilizados en el modelo de Regresión Logística son: solvers, penalty y c values.

Hiperparámetro	Valor
solver	liblinear
penalty	l2
c values	10

Tabla 7.3: Tabla de Hiperparámetros Modelo Regresión Logística Primer Ciclo

Los hiperparámetros utilizados en el modelo de Red Neuronal son los que muestra la Tabla 7.3

7.2.2 *Composición Segundo Ciclo*

En el segundo ciclo, se modificó el conjunto de datos utilizados como datos de entrada para los modelos. Los cambios fueron los siguientes:

1. Eliminación de variables Open, High, Low y Close del conjunto de datos utilizado
2. Creación de características lineales y autorregresivas
3. Desplazamiento de 1 instancia al momento de crear las características para únicamente utilizar información del pasado

Adicional se modificaron los valores de los hiperparámetros utilizados y la cantidad de ellos.

Hiperparámetro	Valor
hidden layers sizes	500,500
activation	tahn
solver	Adam
alpha	0.0001
batch size	100m
learning rate	constant

Tabla 7.4: Tabla de Hiperparámetros Modelo Red Neuronal Segundo Ciclo

Los hiperparámetros utilizados en el modelo de Perceptrón Multicapa son los que muestra la Tabla 7.4

Hiperparámetro	Valor
solver	liblinear
penalty	l1
c values	1

Tabla 7.5: Tabla de Hiperparámetros Modelo Regresión Logística Segundo Ciclo

Los hiperparámetros utilizados en el modelo de Regresión Logística son los que muestra la Tabla 7.4

7.2.3 *Tiempos de Búsqueda de Hiperparámetros, Entrenamiento y Prueba*

Tiempos Primer y Segundo Ciclo				
Tiempo / Modelo	Red Neuronal	Regresión Logística	Perceptrón Multicapa	Regresión Logística
Hiperparámetros	49 minutos	12 minutos	1 h 23 minutos	14 minutos
Entrenamiento	25 segundos	9 segundos	1 min	16 segundos
Prueba	1 segundo	1 segundo	1 segundo	1 segundo

Tabla 7.6: Tabla de Tiempos obtenidos en el Primer y Segundo Ciclo

Existen diferencias entre la ejecución del Primer Ciclo y el Segundo. Tomando en cuenta la Tabla 7.6, los valores más grandes en cuestión de tiempo son utilizados para la búsqueda de hiperparámetros. El valor del Perceptrón Multicapa en el Segundo Ciclo es casi el doble que en la Red Neuronal del Primer Ciclo. Comparando la misma categoría pero diferente modelo, hubo un aumento de 2 minutos entre el Primer y Segundo Ciclo con el modelo de Regresión Logística.

Los tiempos de entrenamiento y prueba son significativamente menores en todos los casos a comparación de la búsqueda de hiperparámetros. Existe un aumento en la categoría de entrenamiento del doble de tiempo comparando el Primer y Segundo Ciclo. En la categoría de prueba, no existen diferencias significativas entre el Primer y Segundo Ciclo.

7.2.4 *Importancia de las Características*

La hipótesis, expresada en el Capítulo 6, habla sobre la importancia del Volumen. En la siguiente lista se comparten las primeras 10 características en orden descendente de acuerdo a la importancia:

1. ho
2. ol
3. hl
4. ma hl 2
5. ma ol 2
6. ma hl 2
7. ma hl 3
8. Quote asset volume
9. ma ol 3
10. Taker buy base asset volume

Las primeras características son las diferencias entre precios, High-Open, Open-Low y High-Low. Las siguientes cuatro son promedios móviles de las características antes mencionadas. En el lugar 8 y 10 hay dos características relacionadas con volumen: Quote asset volume y Taker buy base asset volume.

En conclusión, el volumen si forma parte importante para el cambio de precio, no está dentro de las 5 características principales utilizadas pero si dentro de las 10 principales. La manera en la que interactúa el volumen es desde 2 perspectivas: volumen total del activo en el periodo de tiempo (1 hora) y volumen total del activo base comprado durante el periodo de tiempo (1 hora).

8 Prototipo

En este capítulo se presenta materialmente lo que se logra con el trabajo.

Este proyecto es un trabajo de exploración. A continuación se presenta una tabla con las características y su descripción:

Característica	Descripción
Tipo de Modelo	Perceptrón Multicapa, Regresión Logística
Objetivo del Modelo	Predecir cambios de signo de la variable objetivo
Datos	Características Lineales, Autorregresivas
Entradas	7,293, 41 características
Salidas	Binario, cambio de signo
% Subconjunto de Entrenamiento	80
% Subconjunto de Prueba	20
Preprocesamiento	Escalamiento con MinMaxScaler
Evaluación del Modelo	AUC-ROC, exactitud, precisión, sensibilidad, Valor F1

Tabla 8.1: Tabla Descriptiva de Prototipo

Este trabajo de exploración usa 2 modelos distintos: Perceptrón Multicapa y Regresión Logística. Estos comparten las mismas entradas y se evaluan de la misma manera. Como ya se mencionó en el capítulo 4, el objetivo es desarrollar un proceso de modelado predictivo para pronósticar movimientos del mercado. En la Tabla 8.1 se exponen los tipos de modelos, objetivo de ellos, datos, entradas, salidas, división de datos de entrenamiento y prueba, preprocesamiento y las métricas de evaluación.

9 Trabajo Futuro

En este capítulo se presentan las intenciones de continuar con el proyecto y los aspectos en los que se haría. Se divide en dos secciones: Alcance y Limitaciones y Resultado y Extensión Futuro.

9.1 Alcance y Limitaciones

Es importante mencionar cual es el alcance y que límites tiene este proyecto. Primero, el objetivo principal es el desarrollo de un proceso de modelado predictivo para pronósticar el cambio de signo de la variable objetivo. El conjunto de datos utilizado es meramente cuantitativo y no toma en cuenta los aspectos sociales, políticos y económicos que sucedieron en el periodo de tiempo elegido. Por último, se busca comprobar la hipótesis de que el volumen es una variable importante que determina el cambio de precio.

9.2 Resultado y Extensión Futuro

Basado en lo que escribe Stefan Jansen en su libro *Machine Learning for Algorithmic Trading*[1], como extensión futuro se buscaría implementar y reproducir un agente de trading utilizando *Deep Reinforcement Learning*.

[1] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020

El aprendizaje mediante refuerzo toma una perspectiva diferente que los algoritmos supervisados y no supervisados. Con esto y con los comentarios que Jansen expone en su libro[2] que expresa que este tipo de aprendizaje se adapta bastante bien al sector financiero.

[2] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020

10 Apéndice A

En este capítulo se presentan los sesgos e implicaciones del proyecto. Se divide en dos secciones: Sesgos e Implicaciones.

10.1 Sesgos

En este apartado se exponen 2 subsecciones para el ánalisis del sesgo: Datos de Entrada y Selección de Características

10.1.1 Sesgos en los Datos de Entrada

Como parte del estudio se tomaron los datos del 1 de enero del 2023 hasta el 31 de octubre del mismo año. Sin embargo esto pudiera generar sesgo en las predicciones ya que pudiera ser el caso que no sea representativa la información recolectada.

Existe la posibilidad que en el periodo de datos recolectado no existan acontecimientos comunes en el cambio de precio y por eso mismo no se puede reproducir ese patrón. De la misma manera, puede acontecer lo opuesto, que en el periodo de tiempo escogido haya acontecido algo fuera de lo común y por ello no pueda predecir de manera correcta.

10.1.2 Sesgos en la Selección de Características

Se crearon características lineales y autorregresivas, las lineales no parecen alarmar la ejecución y los resultados. Sin embargo, las autorregresivas pudieran afectar los resultados por los rezagos analizados. En la creación de las características se utilizan 2 y 3 rezagos pero pudiera ser el caso que al aumentar ese número exista un comportamiento distinto.

10.2 Implicaciones

En este apartado se abordan las implicaciones desde 2 perpectivas: Riesgo de Decisiones Incorrectas y Necesidad de Transparencia y Ética.

10.2.1 Riesgo de Decisiones Incorrectas

Los resultados obtenidos de los modelos de Machine Learning pueden influir en la decisión financiera, sin embargo es importante recalcar que estas predicciones son probabilísticas, existe la posibilidad de error. Es por ello que el usuario debe utilizar estos resultados como herramienta complementaria, no como verdad absoluta.

10.2.2 Necesidad de Transparencia y Ética

Tal y como se menciona en la subsección anterior, los resultados de los modelos son una herramienta para la toma de decisiones. Es necesario que la información sea verdadera, sin sesgo para poder tomar decisiones basadas en datos de la realidad, si hay modificaciones o alteraciones en la información, los resultados no tendrán sentido.

Es fácil alterar la información para aconsejar o que otro tercero tome decisiones incorrectas. Es importante tener los datos necesarios para poder utilizar esta herramienta de manera correcta y sobre todo, desde el punto de vista ético, los datos utilizados, no deberán tener ningún interés contrapuesto ya sea personal, financiero, sociopolítico o cualquier interés ajeno a reflejar lo que sucede en la realidad.

11 Apéndice B

En este capítulo se proporcionan instrucciones sobre la implementación y uso del prototipo desarrollado. El objetivo es facilitar la comprensión y utilización de lo que se desarrolló en este proyecto.

11.1 Repositorio en GitHub y Licencia

El código y recursos relacionados a este trabajo están diponibles en el repositorio de GitHub.

Este trabajo se distribuye bajo la licencia **GNU General Public License V3**, que establece los términos y condiciones para su uso y distribución.

11.2 Ejecución del Prototipo

A continuación se describe el proceso para reproducir el prototipo expuesto en este trabajo:

11.2.1 Librerías Necesarias

- Pandas
- Numpy
- datetime
- seaborn
- matplotlib.pyplot
- sklearn
- plotly.graph_objects

11.2.2 Importación de Datos

1. Acceder al repositorio de GitHub para descargar los datos o replicar lo que se expone en el Capítulo 6.
2. Importar los datos como dataframe.

11.2.3 *Análisis Descriptivo y Exploratorio*

- Función para Análisis Descriptivo
- Función para Análisis Exploratorio
- Función para Ajuste de Distribución de Probabilidad Empírica

11.2.4 *Ingeniería de Características y Preprocesamiento*

- Función para Características Lineales
- Función para Características Autorregresivas
- Función para Preprocesamiento

11.2.5 *Modelos Utilizados*

- Función para Modelo de Regresión Logística
- Función para Modelo de Perceptrón Multicapa

12 Apéndice C

En este capítulo se encuentra contenido de ayuda para la ejecución de las funciones utilizadas en este trabajo.

12.1 Funciones

12.1.1 Función Análisis Descriptivo

```
def procesar_dataframe(df):
  # Agregar nombres a las columnas
  df.columns = ['Open_time','Open','High','Low','Close',
                'Volume','Close_time','Quote_asset_volume',
                'Number_of_trades','Taker_buy_base_asset_volume',
                'Taker_buy_quote_asset_volume','Ignore','Response']

  # Mostrar informaci n
  print(df.info())

  # Mostrar estad stica descriptiva
  print(df.describe())

  # Convertir las columnas de 'Open_time' y 'Close_time' a formato de fecha y hora
  df['Open_time'] = pd.to_datetime(df['Open_time'], unit='ms', origin='unix')
  df['Close_time'] = pd.to_datetime(df['Close_time'], unit='ms', origin='unix')

  # Mostrar el dataframe
  print(df.head())

  return df
```

12.1.2 Función Análisis Exploratorio

```
def analisis_exploratorio(df):
    # Imprimir histogramas de todas las variables
    df.hist()
    plt.show()

    # Calcular simetr a de las variables
    simetria = df.skew()
    print("Simetr a de las variables:")
    print(simetria)

    # Calcular curtosis de las variables
    curtosis = df.kurt()
    print("\nCurtosis de las variables:")
    print(curtosis)

    # Dividir el tiempo en d a de la semana, hora del d a y mes
    df['Open_time'] = pd.to_datetime(df['Open_time'], unit='ms', origin='unix')
    df.set_index('Open_time', inplace=True)
    df['day_of_week'] = df.index.dayofweek
    df['hour_of_day'] = df.index.hour
```

```
    df['month'] = df.index.month

    # Histograma de d a de la semana
    sns.countplot(x='day_of_week', data=df)
    plt.xlabel('D a de la Semana')
    plt.ylabel('Frecuencia')
    plt.show()

    # Histograma de hora del d a
    sns.countplot(x='hour_of_day', data=df)
    plt.xlabel('Hora del D a')
    plt.ylabel('Frecuencia')
    plt.show()

    # Histograma de mes
    sns.countplot(x='month', data=df)
    plt.xlabel('Mes')
    plt.ylabel('Frecuencia')
    plt.show()

    # Calcular valores at picos
    q1 = df.quantile(0.25)
    q3 = df.quantile(0.75)
    iqr = q3 - q1
    print("\nValores at picos (IQR):")
    print(iqr)

    # Ajuste de distribuci n de probabilidad emp rica

    # Boxplot de precios
    sns.boxplot(data=df[['Open', 'High', 'Low', 'Close']])
    plt.show()

    # Boxplot de volumen y n mero de operaciones
    sns.boxplot(data=df[['Volume', 'Number_of_trades']])
    plt.show()

    # Boxplot de volumen y volumen de compra
    sns.boxplot(data=df[['Quote_asset_volume', 'Taker_buy_quote_asset_volume']])
    plt.show()

    # Boxplot de volumen de compra de activos base
    sns.boxplot(data=df['Taker_buy_base_asset_volume'])
    plt.show()
```

12.1.3 *Función Ajuste de Distribución de Probabilidad Empírica*

```
def ajuste_distribuciones_empiricas(df, columnas):
    # Iterar sobre las columnas especificadas
    for columna in columnas:
        datos = df[columna]

        # Obtener los par metros de ajuste para varias distribuciones
        distribuciones = ['gamma', 'expon', 'lognorm', 'beta', 'norm']
        resultados_evaluacion = []
        mejor_distribucion = None
        mejor_parametros = None
        mejor_sse = float('inf')

        for distribucion in distribuciones:
            # Ajustar la distribuci n a los datos
            distribucion_actual = getattr(stats, distribucion)
            parametros = distribucion_actual.fit(datos)

            # Calcular el error cuadr tico medio (SSE) para evaluar el ajuste
            sse = sum((distribucion_actual.pdf(datos, *parametros) - datos)**2)

            # Calcular el AIC y el BIC
            n = len(datos)
            k = len(parametros)
            aic = n * np.log(sse / n) + 2 * k
            bic = n * np.log(sse / n) + k * np.log(n)

            resultados_evaluacion.append({
                'Distribuci n': distribucion,
```

```
                'SSE': sse,
                'AIC': aic,
                'BIC': bic
            })
            # Actualizar la mejor distribuci n si encontramos un SSE m s bajo
            if sse < mejor_sse:
                mejor_distribucion = distribucion_actual
                mejor_parametros = parametros
                mejor_sse = sse

        df_resultados = pd.DataFrame(resultados_evaluacion)
        # Configuraci n del estilo de la gr fica
        sns.set(style="whitegrid")

        # Creaci n de la gr fica con histograma y distribuci n mejor ajustada
        plt.figure(figsize=(8, 6))
        sns.histplot(datos, bins=30, kde=True, color='skyblue', label='Datos', stat='density')
        x = np.linspace(datos.min(), datos.max(), 100)
        pdf = mejor_distribucion.pdf(x, *mejor_parametros)
        plt.plot(x, pdf, 'r-', linewidth=2, label='Distribuci n Mejor Ajustada')
        plt.xlabel(columna)
        plt.ylabel('Densidad de Probabilidad')
        plt.title(f'Ajuste de Distribuci n a {columna}')
        plt.legend()
        plt.show()

        print(df_resultados)
```

12.1.4 *Función Creación Características Lineales*

```
def calcular_caracteristicas_lineales(df):
    # Calcular caracter sticas lineales
    df['co'] = df['Close'].shift(-1) - df['Open']
    df['hl'] = df['High'].shift(-1) - df['Low']
    df['ol'] = df['Open'].shift(-1) - df['Low']
    df['ho'] = df['High'].shift(-1) - df['Open']
    df['cov'] = df['co'] / df['Volume']
    df['hlv'] = df['hl'] / df['Volume']

    # Eliminar filas con valores faltantes
    df.dropna(inplace=True)

    # Eliminar columnas no deseadas
    df.drop(['Open', 'High', 'Low', 'Close', 'Ignore'], axis=1, inplace=True)

    # Contar valores faltantes
    missing_values_count = df.isnull().sum()
    print(missing_values_count)

    return df
```

12.1.5 *Función Creación Características Autorregresivas*

```
def calcular_caracteristicas_autorregresivas(df):
    # Calcular caracter sticas autorregresivas con ventana 2
    df['ma_ol_2'] = df['ol'].rolling(2).mean()
    df['ma_ho_2'] = df['ho'].rolling(2).mean()
    df['ma_hl_2'] = df['hl'].rolling(2).mean()
    df['ma_hlv_2'] = df['hlv'].rolling(2).mean()
    df['ma_cov_2'] = df['cov'].rolling(2).mean()

    df['sd_ol_2'] = df['ol'].rolling(2).std()
    df['sd_ho_2'] = df['ho'].rolling(2).std()
    df['sd_hl_2'] = df['hl'].rolling(2).std()
    df['sd_hlv_2'] = df['hlv'].rolling(2).std()
    df['sd_cov_2'] = df['cov'].rolling(2).std()

    df['lag_vol_2'] = df['Volume'].shift(2)
    df['sum_vol_2'] = df['Volume'].rolling(2).sum()
    df['mean_vol_2'] = df['Volume'].rolling(2).mean()

```

```
    # Calcular caracter sticas autorregresivas con ventana 3
    df['ma_ol_3'] = df['ol'].rolling(3).mean()
    df['ma_ho_3'] = df['ho'].rolling(3).mean()
    df['ma_hl_3'] = df['hl'].rolling(3).mean()
    df['ma_hlv_3'] = df['hlv'].rolling(3).mean()
    df['ma_cov_3'] = df['cov'].rolling(3).mean()

    df['sd_ol_3'] = df['ol'].rolling(3).std()
    df['sd_ho_3'] = df['ho'].rolling(3).std()
    df['sd_hl_3'] = df['hl'].rolling(3).std()
    df['sd_hlv_3'] = df['hlv'].rolling(3).std()
    df['sd_cov_3'] = df['cov'].rolling(3).std()

    df['lag_vol_3'] = df['Volume'].shift(3)
    df['sum_vol_3'] = df['Volume'].rolling(3).sum()
    df['mean_vol_3'] = df['Volume'].rolling(3).mean()

    # Llenar valores faltantes con 0
    df_filled = df.fillna(0)

    # Contar valores faltantes
    missing_values_count = df_filled.isnull().sum()
    print(missing_values_count)

    return df_filled
```

12.1.6 *Función para Preprocesamiento*

```
def preprocesamiento(df):
    # Llenar valores faltantes con 0
    df_filled = df.fillna(0)

    # Crear una copia del DataFrame
    new_df = df_filled.copy()

    # Separar caracter sticas y variable objetivo
    X = new_df.drop(['Response'], axis=1)
    y = new_df['Response']

    # Reemplazar infinitos con 0
    X[np.isinf(X)] = 0

    # Escalar caracter sticas
    scaler = preprocessing.MinMaxScaler()
    X_scaled = scaler.fit_transform(X)

    # Dividir datos en conjuntos de entrenamiento y prueba
    X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=1)

    return X_train, X_test, y_train, y_test
```

12.1.7 *Función Modelo Regresión Logística*

```
def logistic_regression_model(X_train, X_test, y_train, y_test):
    # Definir el modelo de regresi n log stica
    lrmodel = LogisticRegression()

    # Definir los hiperpar metros para la b squeda en cuadr cula
    solvers = ['newton-cg', 'liblinear', 'lbfgs']
    penalty = ['l2', 'l1', 'elasticnet']
    c_values = [100, 10, 1.0, 0.1, 0.01]
    grid = dict(solver=solvers, penalty=penalty, C=c_values)

    # Definir la validaci n cruzada
    cv = RepeatedStratifiedKFold(n_splits=5, n_repeats=5, random_state=1)

    # Realizar la b squeda en cuadr cula
    grid_search = GridSearchCV(estimator=lrmodel, param_grid=grid, n_jobs=-1, cv=3, scoring='accuracy', error_score
      =0)
    grid_result = grid_search.fit(X_train, y_train)

```

```
    # Imprimir los mejores resultados de la b squeda en cuadr cula
    print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))

    # Crear el modelo final con los mejores hiperpar metros encontrados
    logreg = LogisticRegression(**grid_result.best_params_, multi_class='ovr')
    logreg.fit(X_train, y_train)

    # Predecir las etiquetas en el conjunto de prueba
    y_pred_lr = logreg.predict(X_test)

    # Calcular m tricas de evaluaci n
    accuracy_lr = accuracy_score(y_test, y_pred_lr)
    precision_lr = precision_score(y_test, y_pred_lr)
    recall_lr = recall_score(y_test, y_pred_lr)
    f1_score_lr = f1_score(y_test, y_pred_lr)
    conf_matrix = confusion_matrix(y_test, y_pred_lr)

    print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(accuracy_lr))
    print('Precision of logistic regression classifier on test set: {:.2f}'.format(precision_lr))
    print('Recall of logistic regression classifier on test set: {:.2f}'.format(recall_lr))
    print('F1 Score of logistic regression classifier on test set: {:.2f}'.format(f1_score_lr))
    print('Confusion Matrix:')
    print(conf_matrix)

    # Calcular la curva ROC y el  rea  bajo la curva (AUC)
    y_scores_lr = logreg.predict_proba(X_test)[:, 1]
    fpr_lr, tpr_lr, thresholds_lr = roc_curve(y_test, y_scores_lr)
    auc_roc_lr = roc_auc_score(y_test, y_scores_lr)

    # Graficar la curva ROC
    plt.figure(figsize=(8, 8))
    plt.plot(fpr_lr, tpr_lr, color='blue', lw=2, label=f'AUC = {auc_roc_lr:.2f}')
    plt.plot([0, 1], [0, 1], color='green', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title('ROC Curve')
    plt.legend(loc="lower right")
    plt.show()

    return logreg, accuracy_lr, precision_lr, recall_lr, f1_score_lr, conf_matrix
```

12.1.8 *Función Modelo Perceptrón Multicapa*

```
def mlp_classifier_model(X_train, X_test, y_train, y_test):
    # Definir el modelo de Perceptr n Multicapa
    mlp = MLPClassifier(random_state=1)

    # Definir los hiperpar metros para la b squeda en cuadr cula
    param_grid = {
        'hidden_layer_sizes': [(50,), (100,), (50, 50), (100, 50)],
        'activation': ['relu', 'tanh'],
        'solver': ['adam', 'sgd'],
        'alpha': [0.0001, 0.001, 0.01],
        'batch_size': [50, 100, 200],
        'learning_rate': ['constant', 'adaptive']
    }

    # Realizar la b squeda en cuadr cula
    grid_search = GridSearchCV(mlp, param_grid, cv=5, scoring='accuracy')
    grid_search.fit(X_train, y_train)

    # Obtener el mejor modelo y sus hiperpar metros
    best_model = grid_search.best_estimator_
    best_params = grid_search.best_params_

    # Evaluar el mejor modelo en el conjunto de prueba
    y_pred_mlp = best_model.predict(X_test)
    accuracy_mlp = accuracy_score(y_test, y_pred_mlp)
    precision_mlp = precision_score(y_test, y_pred_mlp)
    recall_mlp = recall_score(y_test, y_pred_mlp)
    f1_score_mlp = f1_score(y_test, y_pred_mlp)
    conf_matrix_mlp = confusion_matrix(y_test, y_pred_mlp)

```

```
    print('Accuracy of MLP classifier on test set: {:.2f}'.format(accuracy_mlp))
    print('Precision of MLP classifier on test set: {:.2f}'.format(precision_mlp))
    print('Recall of MLP classifier on test set: {:.2f}'.format(recall_mlp))
    print('F1 Score of MLP classifier on test set: {:.2f}'.format(f1_score_mlp))
    print('Confusion Matrix:')
    print(conf_matrix_mlp)

    # Imprimir los mejores hiperparametros y la precision en el conjunto de prueba
    print("Mejores hiperparametros:", best_params)
    print("Precision en conjunto de prueba:", accuracy_mlp)

    # Calcular la curva ROC y el  rea  bajo la curva (AUC)
    y_scores_mlp = mlp.predict_proba(X_test)[:, 1]
    fpr_mlp, tpr_mlp, thresholdsmlp = roc_curve(y_test, y_scores_mlp)
    auc_roc_mlp = roc_auc_score(y_test, y_scores_mlp)

    # Graficar la curva ROC
    plt.figure(figsize=(8, 8))
    plt.plot(fpr_mlp, tpr_mlp, color='blue', lw=2, label=f'AUC = {auc_roc_mlp:.2f}')
    plt.plot([0, 1], [0, 1], color='green', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title('ROC Curve')
    plt.legend(loc="lower right")
    plt.show()

    # Retornar el mejor modelo, su precision y las predicciones
    return best_model, accuracy_mlp, y_pred_mlp
```

Bibliografía

[1] E. H. Digital, "México se presenta como la futura capital del trading en latinoamérica," 2023. https://heraldodemexico.com.mx/nacional/2023/9/22/mexico-se-presenta-como-la-futura-capital-del-trading-en-latinoamerica-540848.html [Accessed: (03/05/2024)].

[2] M. de Jesús Torres Barrón, "Pronósticos, una herramienta clave para la planeación de las empresas," *Instituto Tecnológico de Sonora*, 2022.

[3] D. Spiegelhalter, *The Art of Statistics How to Learn from Data*. Basic Books, 1st ed., March 2021.

[4] M. López de Prado, *Advances in Financial Machine Learning*. John Wiley Sons, Inc., 1st ed., February 2018.

[5] J. Stefan, *Machine Learning for Algorithmic Trading*. Packt, 2nd ed., July 2020.

[6] T. View, "Ethusd," 2024. https://es.tradingview.com/symbols/ETHUSD/ [Accessed: (03/05/2024)].

[7] A. P. F., "Evaluación de modelos de redes neuronales de predicción del signo de la varaición del ipsa," *Universidad de Chile*, 2002.

[8] A. M. Larrea, "Aplicación de técnicas cuantitativas en el análisis de cotizaciones en mercados energéticos," 2022.

[9] Binance, "Binance faq," 2024. https://www.binance.com/es-MX [Accessed: (26/03/2024)].

[10] J. D. Muñoz-Elguezábal, J. F. Sánchez-Torres, "T-fold sequential-validation technique for out-of-distribution generalization with financial time series data. international conference on econometrics and statistics," 2021.