Predicción del movimiento de la acción de bolsa "S & P 500"

María Luisa Argáez Salcido[a]

[a] *Universidad Autónoma de Nuevo León, Pedro de Alba, Niños Héroes, Ciudad Universitaria, San Nicolás de los Garza, 66451, Nuevo León, México*

Abstract

El presente artículo tiene como objetivo predecir si es conveniente comprar o no en la acción de "S & P 500". Esto se realizó mediante el análisis de series temporales, la generación de nuevas características a partir de los datos de los máximos, mínimos, de la apertura,y de cierre de la acción de cada día en un periodo de tiempo de 20 años. La predicción se realizó utilizando algoritmos de aprendizaje máquina como bosques aleatorios y XGBoost obteniendo resultados de 54 % de exactitud.

Keywords: clasificación, "S & P 500", aprendizaje automático

1. Introducción

1.1. Trading

El comercio de acciones o también llamado en inglés *"Trading"* es la compra y venta de valores, como acciones, bonos, divisas y materias primas, en dónde el éxito comercial depende de la capacidad del comerciante para ser rentable con el tiempo [1].

Ahora bien, el comercio algoritmico, también llamado comúnmente *trading* algorítmico, es el uso de un sistema automatizado para que se realicen operaciones, que se ejecutan de forma predeterminada a través de un algoritmo específicamente sin intervención humana [1].

La ejecución automatizada es el proceso de permitir que la estrategia genere automáticamente señales de ejecución que se envían al corredor sin intervención humana. Esta es la forma más pura de estrategia comercial algorítmica, ya que minimiza los problemas debido a los errores comentidos por naturaleza en la intervención humana.

Es importante para una ejecución automátizada definir los siguientes conceptos:

- Latencia. Se define como el intervalo de tiempo entre una simulación y una respuesta
- Elección del lenguaje de programación. Puede ser $C++$, C y Java o bien MATLAB, R y Python
- Tipo de colocación. Un comerciante minorista probablemente ejecutará su estrategia desde casa durante el horario de mercado en dónde encenderá su computadora, se conectará a la bolsa, actualizará su software de mercado y luego permitirá que el algoritmo se ejecute automáticamente durante el día. Por el contrario, un fondo cuantitativo profesional con importantes activos bajo administración tendrá una infraestructura de servidor ubicada en el intercambio dedicado para reducir la latencia en la medida de lo posible para ejecutar sus estrategias de alta velocidad.

Es fundamental definir la estrategia que se seguirá, por tanto, la identificación de estrategias tiene tanto que ver con las preferencias personales como con el rendimiento de la estrategia, cómo determinar el tipo y la cantidad de datos históricos para la prueba, cómo evaluar desapasionadamente una estrategia comercial y, finalmente, cómo proceder hacia la fase de backtesting y la implementación de la estrategia. Entre las ideas principales a evaluar para una estrategia de abastecimiento se encuentran:

- Investigación de ideas comerciales algorítmicas. Esta característica se basa en comparar su trabajo con otros.
- Evaluación de estrategias comerciales. Una evaluación de estrategia comprende diferentes temas importantes a destacar, en primer lugar la metodología, que involucra ciertos indicadores como la relación de Sharpe, el apalancamiento, volatilidad, la ganancia y perdida, tipo de pérdida, líquidez, parámetros y el punto de referencia. A continuación se entra en detalle en cada uno.
- Metodología
    - Relación de Sharpe. El ratio de Sharpe se caracteriza heurísticamente como la proporción del riesgo/beneficio de la estrategia ya

que cuantifica la magnitud del retorno que se puede lograr para el nivel de volatilidad soportado por la curva de acciones. Naturalmente, se necesita determinar el período y la frecuencia en que estos rendimientos y la volatilidad (desviación estándar).

- Apalancamiento. Ayuda a responder a la pregunta ¿La estrategia requiere un apalancamiento significativo para ser rentable?
- Frecuencia. La frecuencia de la estrategia está relacionada con la tecnología con la que cuenta, el índice de Sharpe y el nivel general de los costos de transacción.
- Volatilidad. La volatilidad está fuertemente relacionada con el "riesgo"de la estrategia. Una mayor volatilidad de las clases de activos subyacentes, si no están cubiertas, a menudo conduce a una mayor volatilidad en la curva de acciones y, por lo tanto, índices de Sharpe más bajos
- Ganancia/pérdida y ganancia/pérdida promedio. Las estrategias diferirán en sus características de ganancia/pérdida y ganancia/pérdida promedio. Uno puede tener una estrategia muy rentable, incluso si el número de operaciones perdedoras supera el número de operaciones ganadoras.Por ejemplo, las estrategias de impulso tienden a tener este patrón, ya que se basan en una pequeña cantidad de "grandes éxitos"para ser rentables. Por otro lado, las estrategias de reversión a la media tienden a tener perfiles opuestos donde la mayoría de las operaciones son "ganadoras", pero las operaciones perdedoras pueden ser bastante graves.
- Reducción máxima. La reducción máxima es la mayor caída porcentual general de pico a valle en la curva de capital de la estrategia. Es bien sabido que las estrategias de impulso sufren períodos de caídas prolongadas (debido a una serie de muchas operaciones perdedoras incrementales). Muchos comerciantes se pueden dar por vencidos en períodos de reducción prolongada, incluso si las pruebas históricas han sugerido que esto es "negocios como de costumbre"para la estrategia. Es importantísimo determinar qué porcentaje de reducción y el período de tiempo es posible aceptar antes de dejar de operar con su estrategia.
- Capacidad/Liquidez. La capacidad determina la escalabilidad de la estrategia para aumentar el capital. Muchos de los fondos de

cobertura más grandes sufren importantes problemas de capacidad a medida que sus estrategias aumentan de capital.

- Parámetros. Se recomienda probar y orientar estrategias con la menor cantidad de parámetros posible o asegurarse de tener suficientes cantidades de datos con los que probar sus estrategias para identificar rápidamente si algún parametro aporta a la estrategia o no[1].

1.2. Aprendizaje Automático

Dichos algoritmos que se ejecutan sin intervención humana provienen del aprendizaje automático, el cual es un subcampo de la inteligencia artificial que permite a las computadoras realizar tareas sin ser explicitamente programadas, si no que a través de un entrenamiento sean capaces de realizar cálculos por ellas mismas.

En el aprendizaje automáticos existen diversos enfoques de aprendizaje, este artículo se enfoca principalmente en el supervisado y se distingue por que la matriz de datos tiene características X que describen a las muestras y una variable de respuesta y, de tal forma que se entrena o se le enseña al algoritmo a encontrar relaciones entre la variable de respuesta y las características para posteriormente, predecir la variable y con base a las características de la muestra.

A su vez, el aprendizaje supervisado se divide en dos grupos llamados de clasificación y de regresión. Se le llama de clasificación cuando se predice una variable categorica y cuando se predice una variable continua se le conoce como de regresión.

1.3. Back-testing

El término de comprobación retrospectiva, también conocido en inglés como *back-testing*, es el proceso de creación de una estrategia automatizada, es que su rendimiento se puede determinar en los datos históricos del mercado, que en el mejor de los casos, son representativos de los datos del mercado futuro.

Entre las ventajas del *backtesting* es que permite una valoración histórica de tal manera que ayuda a determinar las propiedades estadísticas (anteriores) de la estrategia, lo que proporciona información sobre si es probable que una estrategia sea rentable en el futuro. Por otra parte permite mejor eficiencia, en el aspecto de que el comercio algorítmico es sustancialmente más

eficiente que un enfoque discrecional ya que con un sistema completamente automatizado, no hay necesidad de que un individuo o equipo esté constantemente monitoreando los mercados para la acción del precio o la entrada de noticias.

Además permite comparar diferentes estrategias y elegir la más conveniente al mismo tiempo que permite frecuencias más altas de toma de decisiones ya que las estrategias que operan a frecuencias más altas en muchos mercados se vuelven posibles en un entorno automatizado. De hecho, algunas de las estrategias comerciales más rentables operan en el dominio de frecuencia ultra alta en los datos del libro de órdenes limitadas.

Sin embargo, el comercio algorítmico tambien tiene desventajas, entre las cuales se encuentran principalmente requerimientos de gran capital puesto que el comercio algorítmico, generalmente, requiere una base de capital mucho más grande que la que se utilizaría para el comercio minorista discrecional, lo cual se debe al hecho de que hay pocos corredores que admiten la ejecución automatizada de operaciones que no requieren grandes cuentas mínimas.

Además que las muestras de datos de un día o conocidos en inglés como *intraday feed* de los minoristas a menudo tienen un precio de entre $ 300 y $500 por mes, mientras que los feeds comerciales tienen un orden de magnitud superior. Sin dejar de lado que se necesita de una buena infraestructura computacional y un equipo especializado de de programación y experiencia cientifica [1].

1.4. Datos

La obtención de datos históricos es muy importante ya que es la materia prima de los algoritmos de aprendizaje automático. Se sugiere que los datos sean significativos del tema, esto incluye gran variedad de datos como tendencias macroeconómicas, como tasas de interés, cifras de inflación, acciones corporativas (dividendos, división de acciones), cuentas corporativas, cifras de ganancias, informes de cosechas, datos meteorológicos, etc. Estos datos a menudo se usan para valorar empresas u otros activos sobre una base fundamental, datos de noticias, precios de activos, de instrumentos financieros, entre otros.

Es fundamental definir la frecuencia de los datos ya que entre más frecuente, mayores serán los costos, los requisitos de almacenamiento y tecnología, esto da pie a la infraestructura de datos que se necesitará ya que el almacena-

miento de datos financieros puede ser complejo y llegar a requerir demasiado esfuerzo.

Para el comerciante minorista algorítmico o el pequeño fondo cuantitativo, toma en cuenta los conjuntos de datos más comunes que son los precios históricos al final del día, o en inglés *End Of the Day, (EOD)* ,o precios intradía para acciones, índices, futuros (principalmente materias primas o renta fija) y divisas (forex).Para fines de este artículo se utilizarán los datos EOD para índices de acciones. Es posible obtener datos financieros de:Yahoo Finanzas,Finanzas de Google,Cuantificauion, entre otros provedores.

Hay tres formas principales de almacenar datos financieros. Todos ellos poseen diversos grados de acceso, rendimiento y capacidades estructurales. Entre ellos se encuentran: almacenamiento de archivos planos, almacenamiento de documentos-NoSQL y sistemas de gestión de bases de datos relacionales [1].

Exaluar la precisión de los datos es fundamental ya que si no son confiables, se pone en riesgo factores muy importantes como grandes cantidades de dinero.

Los datos de precios históricos de los proveedores son propensos a muchas formas de error como:

- El incorrecto manejo de reparto de acciones y ajustes de dividendos, se debe estar absolutamente seguro de que las fórmulas se han implementado correctamente.

- Detección de Picos. Son puntos de precios que superan en gran medida ciertos niveles históricos de volatilidad. Los picos también pueden ser causados por no tener en cuenta las divisiones de acciones cuando ocurren. Los códigos de filtro de picos se utilizan para notificar a los comerciantes de tales situaciones.

- Agregación de los datos de las acciones llamados en inglés *Open, High, Low, Close (OHLC)*, o bien apertura, máximo, mínimo y cierre del precio. Existen datos gratuitos de *OHLC*, como los de Yahoo/Google, son particularmente propensos a situaciones de "agregación de muestra", o en inglés *"ticks*, incorrectos.[en] las que los intercambios más pequeños procesan transacciones pequeñas muy por encima de los precios de intercambio principales del día, lo que conduce a máximos/sobreinflados.

mínimos una vez agregados. Esto es menos un "errorçomo tal, pero más un problema del que hay que tener cuidado.

- Datos faltantes. Los datos faltantes pueden deberse a la falta de transacciones en un período de tiempo particular (común en datos de resolución de segundo/minuto de empresas de pequeña capitalización sin liquidez), por feriados comerciales o simplemente un error en el sistema de intercambio. Los datos faltantes se pueden rellenar (es decir, rellenar con el valor anterior), interpolar (linealmente o de otro modo) o ignorar, según el sistema de comercio [1].

1.5. Diseño de Experimentos

Técnica desarrollada por Ronald Fisher, el Diseño de Experimentos (DoE , por sus siglas en inglés), es una técnica estadística que estudia el efecto de uno o más factores sobre la media de una variable continua.

Las suposiciones que se plantean es:

$$H_0 : \mu_1 = \mu_2 = ... = \mu_k$$

$$H_1 : \mu_i \neq \mu_j$$

En otras palabras, se plantea la hipótesis nula es el supuesto en que la media de los grupo es igual mientras que la hipótesis alternativa es que al menos una media de los grupos es diferente.

Es importante tener en cuenta las condiciones iniciales para aplicar este tipo de ANOVA, los cuales son:

- Independencia. Es decir que las muestras son independientes y aleatorias.

- Distribución normal. Esto indica que los datos de cada grupo deben seguir una distribución normal. Este supuesto en ocasiones puede ser ignorado ya que la técnica del ANOVA es robusta.

- Homoscedasticidad, Misma varianza entre grupos. La varianza entre los grupos debe de asemejarse. Este supuesto puede no ser tan estricto si existe el mismo número de observaciones por grupo [2].

En este análisis se compararon los resultados de predicción clasificación del algoritmo de bosques aleatorios, XG boost y los datos reales. Es importante destacar que los datos no siguen una distribución normal, más bien es binomial, sin embargo este supuesto no será tan extricto para efectos del ejercicio. Por otro lado, el supuesto de Homocedasticidad puede ser laxo en el aspecto de que existe el mismo número de muestras en los gurpos.

1.6. Análisis presente

Por tanto, es posible concluir que estudiar las acciones es una tarea que implica conocimientos profundos de economia, programación y ciencia de datos para lograr una estrategía de un comercio algorítmico eficiente y poder predecir el movimiento del previo de las acciones, en este caso específico el momento adecuado de cuando comprar una acción.

Además, estudiar las acciones del mercado de valores está estrechamente ligado al crecimiento económico de un país y genera grandes inversiones por parte de los inversores y emite acciones de interés público, pronosticar el movimiento de los precios de las acciones y el mercado se vuelve esencial para evitar grandes pérdidas y tomar decisiones relevantes [3].

Hoy en día la tecnología de la información, todavía se la considera una de las aplicaciones más desafiantes de la predicción de series temporales. Sin embargo, las evidencias existentes de esta área aún carecen de suficientes experimentos de comprensión en la mayoría de los mercados en desarrollo, que han ganado más y más atención recientemente.

Esta investigación tiene la intención de llenar analizar la acción de " S& P 500"para dar recomendaciones de cuando comprar o no en la misma. Dicha predicción de decisión de compra se realiza utilizando algoritmos de aprendizaje máquina, especificamente con bosques aleatorios y XGBoot.

La predicción del movimiento del precio del mercado de valores enfrenta dos corrientes importantes, la primera que establece que no es posible predecir los precios de las acciones en función de los datos disponibles. En contraste con la hipótesis de que si es posible predecir el movimiento de las acciones, siempre y cuando los datos se procesen de una manera eficiente. Si bien, creo que es mejor monitorear y descubrir nuevas estrategias que permitan generar conocimiento y tomar mejores decisiones basadas en datos [4].

En este ejercicio se trabajará un análisis descriptivo de las *intraday ticks* del índice de bolsa de "S& P 500.ªl finalizar el día (EOD), de la generación de características utiles para después aplicar el algoritmo de selección de características secuencial o en inglés *Sequential Feature Selection (SFS)* de acuerdo al algoritmo de aprendizaje automático que se pruebe, ya sea el de bosques aleatorios o el de XGBoost. Después, se incluye la técnica de *Back-testing* y finalmente, se comparan los resultados con una matriz de confusión y la evaluación de su reproducibilidad con un diseño de experimentos con el fin de evaluar la predicción de compra en dicha acción.

2. Revisión de literatura

Actualmente se han realizado numerosas investigaciones sobre el comportamiento de las acciones de la bolsa de valores, entre ellos esta [5], quien compara siete algoritmos de aprendizaje automático, los cuales fueron redes neuronales artificiales, K- Vecinos más cercanos, máquinas de soporte vectorial, árboles de decisión, bosques aleatorios, *Bagging* y *Boosting* en dónde analizó cuatro conjuntos de datos de índices bursátiles de inversión de riesgo. El algoritmo de aprendizaje automático que mostró mejor rendimiento fue el bosque aleatorio.

Por otro lado, [4] alienta a emular las decisiones humanas mientras usa algoritmos de aprendizaje máquina, en su investigación compara cuatro algoritmos de aprendizaje máquina, los cuales fueron redes neuronales artificiales, máquina de soporte vectorial, bosques aleatorios y Naive- Bayes utilizando datos de 19 años del índice de la bolsa de valores llamado " TWSE". Ellos utilizan diez parámetros técnicos que reflejan la condición de las acciones y el índice de precios de las acciones para conocer cada uno de estos modelos, además emplean una capa de preparación de datos deterministas de tendencia para convertir cada uno de los valores continuos del indicador técnico en ∓ 1 , lo que indica un probable movimiento hacia arriba o hacia abajo en el futuro, respectivamente.

Los experimentos con datos de valor continuo muestran que las redes neuronales aritificiales tienen un rendimiento más alto y que los experimentos con datos de valores discretos muestran que el bosque aleatorio con el rendimiento más alto. Además, [4] promueve el aprendizaje con datos deterministas de tendencia reflejando que el rendimiento de todos estos modelos

mejora significativamente, alcanzando un 77 % con todos los algoritmos menos con Naive-Bayes.

[3] Propuso un estudio comparativo de varios algoritmos para pronosticar los precios de diferentes acciones. El estudio se amplió desde los algoritmos tradicionales del aprendizaje automático como boques aleatorios, K- vecinos más cercanos, máquinas de soporte vectorial, Naive Bayes, modelos de aprendizaje profundo como redes neuronales en especial las redes neuronales convolucionales, redes neuronales artificiales, memoria a largo plazo, etc. Este estudio también incluye varios otros enfoques, como análisis de sentimientos, análisis de series temporales y algoritmos basados en gráficos, y compara los resultados de estos algoritmos para predecir los precios de las acciones de varias empresas.

3. Marco teórico

En este ejercicio se introduce a los algoritmos de ensamble que tiene como objetivo principal mejorar el rendimiento de los algoritmos de aprendizaje máquina de tal forma que combinan algoritmos de diferentes maneras para lograr un modelo que generalice mejor y presente mejores resultados.

3.1. Métodos de ensamble

Los conceptos de *Bagging* o *Boosting* son las técnicas más populares de ensamble, las cuales ofrecen técnicas distintas para mejorar la exactitud de un algoritmo predictivo.

El aprendizaje por ensamble surge bajo la necesidad de que algunos modelos individuales del aprendizaje automático (AA) es que tienden a funcionar mal, lo cual implica tener baja presición al realizar una predicción. De tal manera que los modelos individuales que se combinan para hacer uno más fuerte, se le conocen como algoritmos débiles, o en inglés *weak learners*, ya que tienen un sesgo alto o una variación alta, razón por la cual no es posible que aprendan de manera eficiente y presentan un mal desempeño.

Cuando un modelo presenta alto sesgo es por que no aprendió suficientemente bien de los datos presentados, lo cual implica que las predicciones no tendrán sentido con los datos de entrada. Por otro lado, un modelo con varianza alta resulta de haber aprendido demasiado bien de los datos de tal manera que presenta predicciones muy diferentes de acuerdo a los datos de entrada y resulta difícil predecir con presición el siguiente punto. La relación

de sesgo-varianza en un modelo de ensamble de bajo rendimiento tiene un sesgo alto y una varianza baja, mientras que un modelo sobreajustado tiene una varianza alta y un sesgo bajo. En cualquier caso, no hay equilibrio entre el sesgo y la varianza. Para que haya un equilibrio, tanto el sesgo como la varianza deben ser bajos. El aprendizaje por ensamble intenta equilibrar este equilibrio entre sesgo y varianza reduciendo el sesgo o la varianza.

El aprendizaje conjunto mejora el rendimiento de un modelo principalmente de tres maneras. La primera es reducir la varianza de los *weak learners*, reducir el sesgo de los *weak learners* y mejorar la precisión general de los algoritmos que son mejores.

Para reducir una alta varianza, se utiliza *Bagging* también conocido como *Bootstrap aggregating* , ya que tiene por objetivo hacer un modelo con baja varianca a diferencia de un modelo individual débil. Cuando se menciona, *Bootstrap*, hace referencia a que se toman subconjuntos de datos del conjunto de datos inicial y a estos subconjuntos de datos se denominan conjuntos de datos de arranque o, simplemente, arranques. Remuestreado 'con reemplazo' significa que un punto de datos individual se puede muestrear varias veces y con cada conjunto de datos de arranque se utiliza para entrenar a un alumno débil. Por otra parte, *aggregating* hace referencia a que los *weak learners* individuales son entrenados independientemente unos de otros y los resultados de esas predicciones se agregan al final para obtener la predicción general ya sea utilizando la votación máxima o el promedio.

En contraste, para reducir un sesgo alto, se utiliza el *Boosting*. Usamos *boosting* para combinar *weak learners*con alto sesgo. El *boosting* tiene como objetivo producir un modelo con un sesgo más bajo que el de los modelos individuales. Al igual que en el *bagging*, los*weak learners* son homogéneos.

El *boosting* implica enseñar secuencialmente a los*weak learners*. Aquí, cada modelo subsiguiente mejora los errores de los modelos anteriores en secuencia. Primero se toma una muestra de datos del conjunto de datos inicial, dicha muestra se utiliza para entrenar el primer modelo y el modelo hace su predicción. Las muestras pueden predecirse correcta o incorrectamente, pero las muestras que se predicen incorrectamente se reutilizan para entrenar el siguiente modelo. De esta forma, los modelos posteriores pueden mejorar los errores de los modelos anteriores.

A diferencia del *bagging* , que agrega los resultados de la predicción al final, el *boosting* agrega los resultados en cada paso y se agregan utilizando un promedio ponderado. El promedio ponderado implica otorgar a todos los modelos diferentes pesos según su poder predictivo. En otras palabras,

da más peso al modelo con mayor poder predictivo. Esto se debe a que el alumno con mayor poder predictivo se considera el más importante [6].

De tal manera que se compararon los resultados de un algoritmo de *bagging* y otro de *boosting*, los cuales son bosques aleatorios y el XGBoost.

3.1.1. Bosques Aleatorios

Los bosques aleatorios es un conjunto basado en árboles aleatorios con cada árbol dependiendo de una colección de variables aleatorias. Más formalmente, para un vector aleatorio p-dimensional $X = (X_1, ..., X_p)^T$ que representa la entrada de valor real y una variable aleatoria Y que representa la respuesta de valor real. Se supone una distribución conjunta desconocida $P_{XY}(X, Y)$. El objetivo es encontrar una función de predicción , $f(X)$ para predecir Y. La función de predicción está determinada por una función de pérdida $L(Y, f(X))$ y definido para minimizar el valor esperado de la pérdida $E_{XY}(L(Y, f(X)))$.

Intuitivamente, $L(Y, f(X))$es una medida de qué tan cerca está $f(X)$ de Y; penaliza los valores de $f(X)$ que están muy lejos de Y. Las opciones típicas de L son pérdida de error al cuadrado, $L(Y, f(X)) = (Y - f(X))^2$ para regresión y pérdida cero-uno para clasificación:

$$L(Y, f(X)) = I(Y \neq f(X) = \begin{cases} 0 \ si \ Y = f(X) \\ 1 \ otro \ caso \end{cases}$$

Si se minimiza $E_{XY}(L(Y, f(X)))$ para la pérdida de error al cuadrado da la expectativa condicional, entonces: $f(x) = argmax_{y \in Y} P(Y = y|X = x)$, lo cual se le conoce también como la regla de Bayes.

Los modelos de ensamble construyen f en términos de una colección de los llamados *weak learners*, $h_1(x), ..., h_j(x)$ y estos *weak learners*base se combinan para dar el "predictor de conjunto" $f(x)$. En la clasificación, los *weak learners* se utiliza la votación:

$$f(x) = argmax_{y \in Y} \sum_{j=1}^{J} I(y = h_j)(x))$$

En árboles aleatorios, el $j-$ésimo *weak learner* es un árbol denominado $h_j(X, \theta_j)$, donde X es una colección de variables aleatorias y las θ_j son independientes para $j = 1, ..., J$ [7].

3.1.2. XGBoost

El algoritmo de XGBoost hace referencia por su nombre en inglés *eXtreme Gradient Boosting*, implementa un algoritmo basado en *weak learners* de incremento de gradiente. El incremento de gradiente es un enfoque en el que se crean nuevos modelos que predicen los residuos o errores de modelos anteriores y luego se adicionan para realizar la predicción final. Se llama incremento de gradiente porque utiliza un algoritmo de descenso de gradiente para minimizar la pérdida al agregar nuevos modelos. El algoritmo minimiza el error de predicción con respecto al gradiente negativo de la función de pérdida, similar a un optimizador de descenso de gradiente convencional. El costo computacional durante el entrenamiento es, por lo tanto, proporcional al tiempo que lleva evaluar los posibles puntos de división para cada característica. Entre las características del XGBoost se mencionan las siguientes:

- Regularización. Se utiliza la regularización como L_1 y L_2 para prevenir el sobreajuste.
- Es capaz de manejar datos faltantes.
- Maneja el algoritmo de bosquejo cuantil ponderado distribuido para manejar de manera efectiva los datos ponderados.
- Cuenta con una estructura de bloques para aprendizaje paralelo.
- Optimiza el espacio disponible en el disco y maximiza su uso cuando se manejan grandes conjuntos de datos que no caben en la memoria

En los algoritmos de aprendizaje automático supervisados se cuenta con atributos o características x y la variable respuesta y y el objetivo es pronosticar $y = f(x)$ donde (xi, Yi) con$i = 1, \ldots, n$,.

Se necesita obtener el valor objetivo Y dada la entrada X usando la mejor función, que produce menos errores o pérdidas. La función de pérdida es conocida como $L(y, f)$, la cual debe de ser lo más cercana a cero que se pueda y diferenciable.

Se define la función $f(x)$ tal que:

$$\hat{f}(x) = argmin_{f(x)} E_{x,y}[L(y, f(x))]$$

Es posible restringir el área de búsqueda por una determinada familia de características $f(x, \theta), \theta \in R_d$. Es apropiado, lo cual simplifica el objetivo, ya que ahora tenemos una optimización de valores de parámetros:

$$\hat{f}(x) = f(x, \hat{\theta})$$

$$\hat{\theta} = argmin_{\theta} E_{x,y}[L(y, f(x))]$$

$$\hat{\theta} = \sum_{i=1}^{M} \hat{\theta}_i$$

$$L_{\hat{\theta}} = \sum_{i=1}^{N} L(y_i, f(x_i, \hat{\theta}))$$

Por tanto se busca mínimizar $L_{\hat{\theta}}$, en dónde la opción más utilizada es el gradiente descendente. El pseudocódigo del gradiente descendente es el siguiente:

Algorithm 1 Gradiente descendente

Require: 1. Aproximación inicial de los parámetros $\hat{\theta} = \hat{\theta}_0$

for $t \leftarrow 1$ to M **do**

2. Calcular la función de pérdida de gradiente $\nabla L_{\theta}(\hat{\theta})$ para la aproximación de $\hat{\theta}$:

$$\nabla L_{\theta}(\hat{\theta}) = \left[\frac{\partial L(y, f(x, \theta))}{\partial \theta} \right]_{\theta = \hat{\theta}}$$

3. Declarar la aproximación iterativa actual $\hat{\theta}_t$ basada en el calculo del gradiente $\hat{\theta}_t \leftarrow -\nabla L_{\theta}(\hat{\theta})$

4. Se actualiza la aproximación de parámetros $\hat{\theta} : \hat{\theta} \leftarrow \hat{\theta} + \hat{\theta}_t = \sum_{t=0}^{t} \hat{\theta}_i$

5. Guardar el resultado de la aproximación $\hat{\theta} : \hat{\theta} \leftarrow= \sum_{t=0}^{t} \hat{\theta}_i$

6. Usar la función encontrada $\hat{f}(x) = f(x, \hat{\theta})$

end for

[8]

4. Metodología

4.1. Base de datos

Los datos llamados "OHLC"se deriva del acrónimo en inglés *Open High Low Close* y en español tienen su traducción como apertura, máximo, mínimo y cierre y describen una forma agregada de datos de la bolsa. Los datos de "OHLCïncluyen 4 tipos de datos: la apertura y el cierre representan el primer

y último nivel de precios durante un intervalo específico. Máximo y mínimo representan el precio más alto y más bajo alcanzado durante ese intervalo. Por lo general se agrega el volumen, el cual es la cantidad total negociada durante ese período, sin embargo, para alcance de este ejercicio se excluirá. Este tipo de dato se observa en diferentes frecuencias de muestreo que van desde 1 segundo hasta 1 día.

El análisis de datos se realizó utilizando el software libre *Python*, el cual es un lenguaje de programación de alto nivel, de propósito general y muy popular.

Python cuenta con multiples librerias para diferentes própositos, entre las cuales se encuentra *yfinance*, la cual hace posible obtener los datos "OHLC" de forma directa de diversos índices de inversión de la bolsa de valores, como "S & P 500". Los datos tienen una frecuencia de muestreo de un día y se obtuvieron 23 años de historia, en un periodo de tiempo del 23 de enero de 2000 al 23 de enero de 2023. Es importante denotar que la bolsa de valores abre de lunes a viernes de 9:30 a.m. a 16:00 p.m. y no abre días festivos, por tanto no se tienen 365 muestras por año, si no alrededor de 250 muestras por año.

4.2. Análisis Descriptivo de Datos (ADD) Iniciales

Para la mejor comprensión del lector, primero se realiza un ADD a los datos de "OHLC", posteriormente se explicará la generación de características a partir de estos con su respectivo ADD.

Los datos tienen un periodo de enero 23 del año 2000 al 2023. Los descriptivos de los datos se muestran en el cuadro 1.

Cuadro 1: Descriptivos de datos OHLC

Descriptivos	Open	High	Low	Close
count	5786.00	5786.00	5786.00	5786.00
mean	160.73	161.69	159.67	160.73
std	104.12	104.71	103.46	104.13
min	51.64	53.20	50.99	51.76
25 %	85.37	85.89	84.83	85.32
50 %	107.88	108.52	106.99	107.74
75 %	214.17	214.91	213.51	213.88
max	469.78	470.52	466.68	468.30

En la figura 1, se muestra como el valor de apertura, cierre, mínimo y máximo estan altamente correlacionados positivamente ya que presentando cantidades muy cercanas a uno.

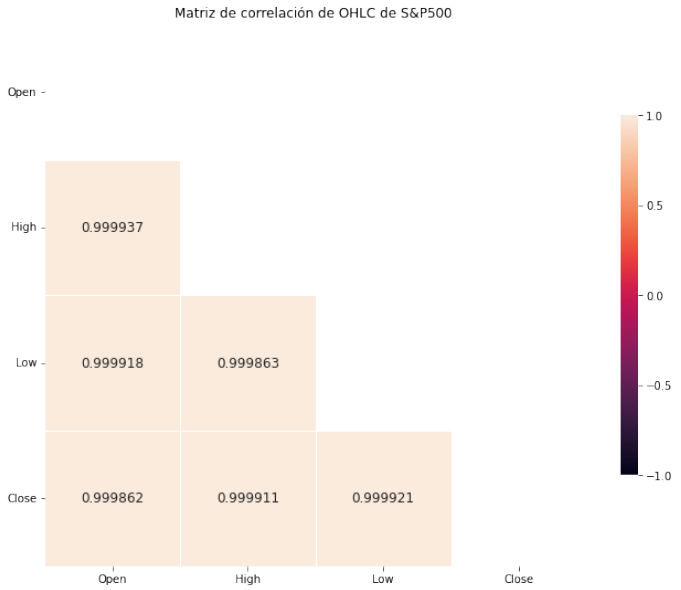

Figura 1: Matriz de correlación de los datos de S&P 500

La figura 2 muestra el cierre de precio desde el año 2000 hasta el 2023. Este artículo se enfocará en el cierre de precio como característica principal.

En la figura 3, se muestra un gráfico de velas que muestra como se ha comportado la acción a lo largo del tiempo.

4.3. Generación de características

A partir de los datos de "OHLCz de la fecha es posible generar nuevas características que aportan información y enriquecen a la base de datos. Las nuevas características se describen a continuación.

- Día. Se extrae el día en que se obtuvo la muestra.
- Mes. Se extrajó el mes en que se obtuvo la muestra.
- Año. Se obtuvo el año en que se obtuvo la muestra.
- Mañana. Se obtuvo el precio de cierre de un día después. Esta nueva columna, es precursora de la variable respuesta de este análisis.

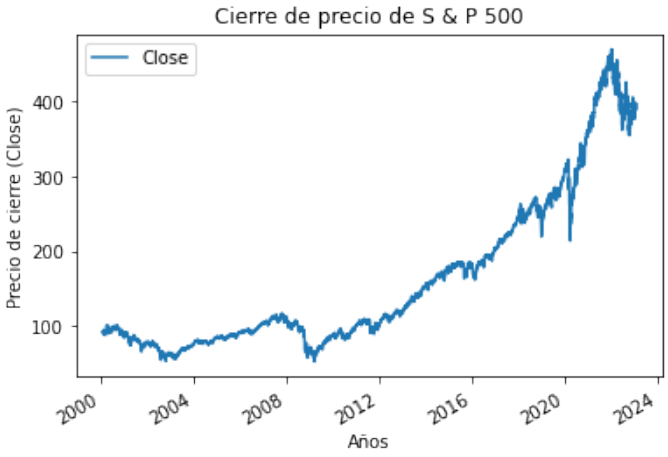

Figura 2: Matriz de correlación de los datos de S&P 500

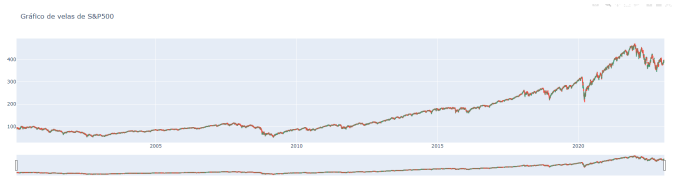

Figura 3: Gráfico de vela de S&P 500

- Objetivo. La variable objetivo, es la variable que se quiere predecir. Los valores son 1 si el valor de la columna "Mañana.^{es} mayor a la columna "Cierre". Es cero en otro caso. Esto indica que si el precio de mañana es mayor al de hoy, es conveniente comprar en la acción.

- Término de trimestre. Esta característica es importante ya que empresas, analistas financieros y agencias gubernamentales publican informes y datos críticos al final de un trimestre y los inversionistas suelen utilizar el final de un trimestre para reevaluar y reequilibrar sus carteras.

- Apertura-Cierre. ES la diferencia entre el valor de apertura y el de cierre. En el mejor de los casos, es negativa ya que quiere decir que el valor de la acción ha subido.

- Mínimo-máximo. Es la diferencia entre el valor mínimo y el máximo y por lo general será positiva.

- Proporción de cierre en 2, 5, 60, 250 y 1000 días. Estas características describen la relación del precio entre la media de los últimos 2, 5,60 ,250 o 1000 días que se tomaron en cuenta.

- Tendencia de cierre en 2, 5, 60, 250 y 1000 días. Es la suma del periodo seleccionado, ya sea a 2, 5, 60 ,250 o 1000 días.

4.4. Análisis Exploratorio de Datos

Ahora que ya se han generado más características que aportan información al análisis, se muestran los gráficos 4 y 5 de distribución y de cajas y bigotes respectivamente, para cada característica. Asi mismo en el cuadro 2 se observan los descriptivos de cada variable.

Entre las conclusiones de los descriptivos se observa que la variable de mañana tiene la mayoria de sus muestras en un rango de 0 a 100. La variable de apertura-cierre se centra en el cero presentando sospecha de una distribución normal. La variable de mínimo-máximo presenta una distribución exponencial. Las variables de las proporciones se aproximan a una distribución normal, sin embargo conforme involucran periodos más grandes su distribución se ve distorsionada. En los diagramas de caja y bigotes en general, se observan gran cantidad de datos atípicos, sin embargo es importante ver que datos atípicos son aquellos que se prefieren y cuales no. Por ejemplo, en la diferencia de mínimo y máximo se quiere que la diferencia sea menor a cero

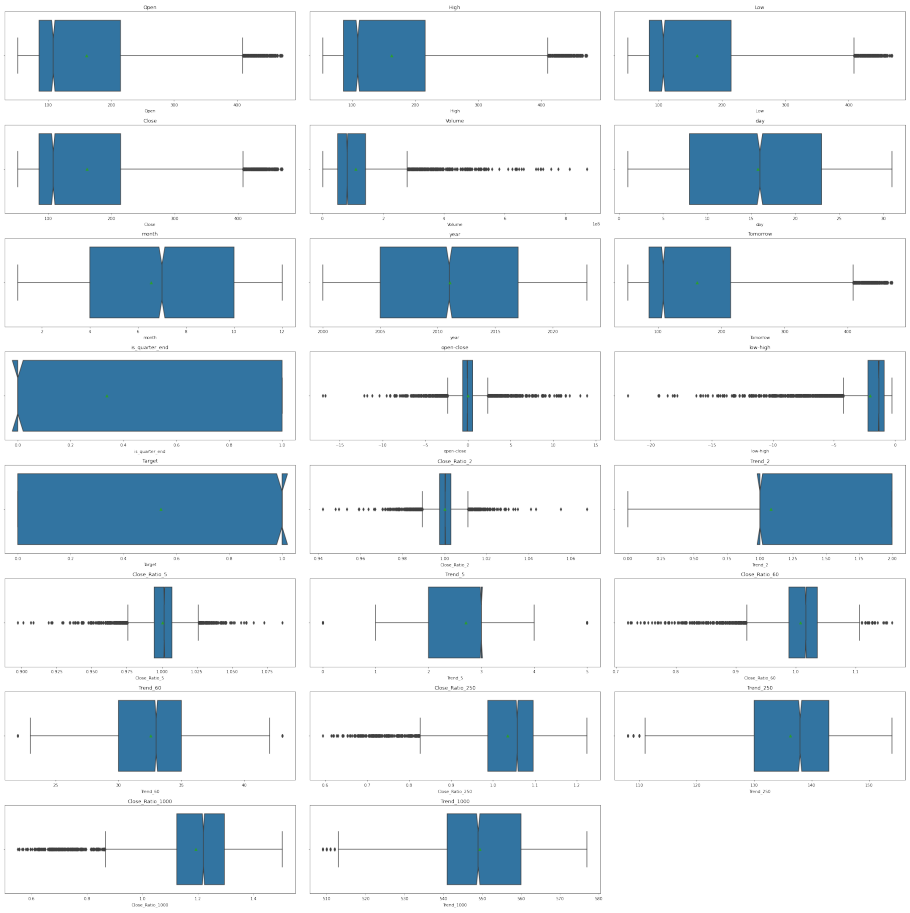

Figura 4: Gráfico de distribución de datos

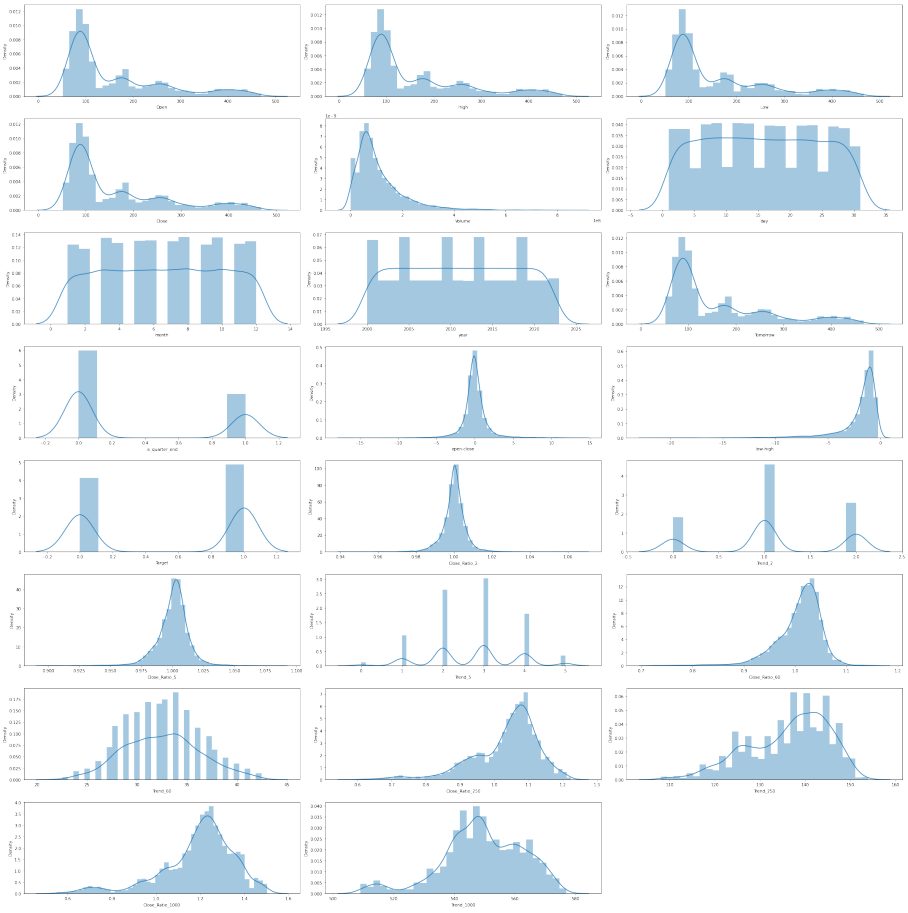

Figura 5: Gráfico de caja y bigotes

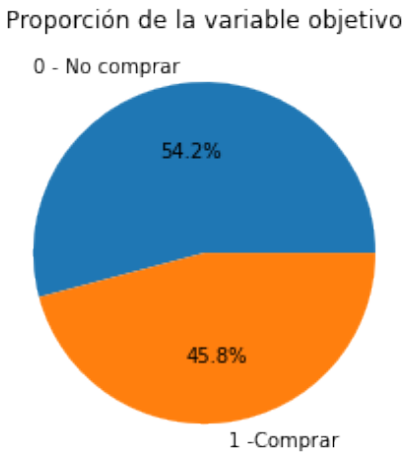

Figura 6: Proporción de la variable objetivo: Decisión de comprar o no en la acción

para que el máximo del cierre sea mucho mayor al valor mínimo presentado en ese día, lo cual indicaría un buen comportamiento de la acción, aun que la mayoria de los datos tienden a cero.

Respecto a las correlaciones que existen en la figura 7, es posible concluir que la variable objetivo presenta correlaciones muy bajas con todas las características. Sin embargo, la variable mañana, presenta correlaciones medias con la proporción de cierre de 60 días,con la variable tendencia a 1000 días y correlaciones altas con año, por la naturaleza del problema se conoce de antemano que se tiene correlación con el mínimo, máximo, apertura y cierre de la acción.

Por otro lado, se conoce que la variable objetivo es categorica y esta conformada por 0 y 1. La proporción de valores nulos y de unidad se refleja en la figura 6.

4.5. Modelos de Aprendizaje Automático

Se utilizaron dos algoritmos de aprendizaje supervisado para clasificación.

Los algoritmos de aprendizaje supervisado, o en inglés *Machine Learning (ML)*, se conocen por seguir una serie de pasos para predecir una variable con base a los datos de entrada. Existen varios procesos de los algortimos de ML entre los cuales estan los de regresión y clasificación, los primeros se caracterizan por predecir una variable continua mientras que los de clasificación predicen una variable categorica. En este análisis se utilizan los algoritmos

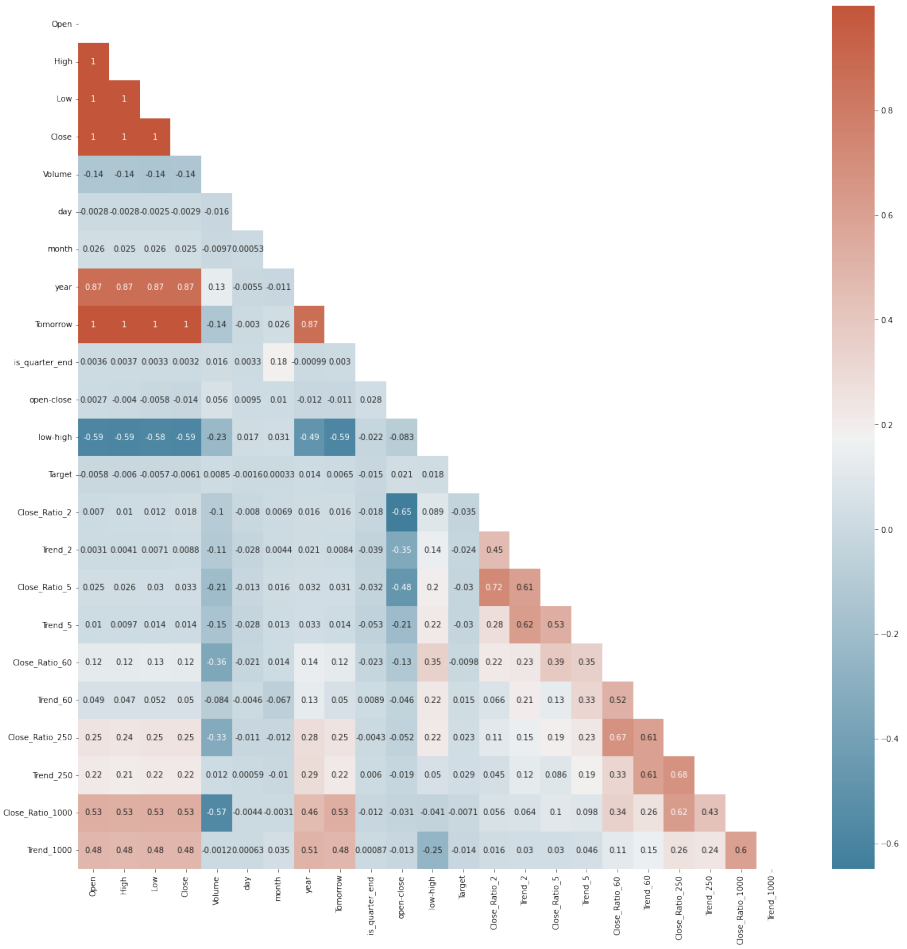

Figura 7: Correlación de características y variable objetivo

Cuadro 2: Descriptivos de las características

Car/Des	Frec	Media	Des.Est.	Mín	25 %	50 %	75 %	Máx
Apertura	5786	160	104.12	51.64	85.37	107.88	214	469
Máximo	5786	161	104.71	53.20	85.89	108.52	214	470
Mínimo	5786	159	103.46	50.99	84.83	106.99	213	466
Cierre	5786	160	104.13	51.76	85.32	107.74	213	468
Día	5786	15.7	8.76	1.00	8.00	16.00	23.0	31.0
Mes	5786	6.54	3.43	1.00	4.00	7.00	10	12.0
Año	5786	2010	6.63	2000.00	2005.00	2011.00	2021	2023
Mañana	5785	160	104.13	51.76	85.32	107.74	213	468
Trimestre	5786	0.340	0.47	0.00	0.00	0.00	1.00	1
Aper-cie	5786	-0.01	1.73	-16.94	-0.61	-0.06	0.550	13.9
Mín-máx	5786	-2.02	2.12	-21.80	-2.20	-1.33	-0.880	-24.0
Objetivo	5786	0.540	0.50	0.00	0.00	1.00	1	1
C_prop_2	5785	1.00	0.01	0.94	1.00	1.00	1	1.07
Tend_2	5784	1.08	0.70	0.00	1.00	1.00	2	2
C_prop_5	5782	1.00	0.01	0.90	0.99	1.00	1	1.09
Tend_5	5781	2.71	1.08	0.00	2.00	3.00	3	5
C_prop_60	5727	1	0.04	0.72	0.99	1.02	1.04	1.16
Tend_60	5726	32.5	3.82	22.00	30.00	33.00	35.0	43.0
C_prop_250	5537	1.04	0.10	0.59	0.99	1.06	1.10	1.22
Tend_250	5536	136	8.90	108.00	130.00	138.00	143	154
C_prop_1000	4787	1.19	0.16	0.55	1.12	1.22	1.3	1.5
Tend_1000	4786	549	13.40	509.00	541.00	549.00	560	570

de aprendizaje máquina supervisados del tipo de clasificación, de tal manera que se predice la variable objetivo, la cual describe si el precio de cierre de hoy es menor al de mañana para tomar la decisión de si comprar o no en la acción.

Existe gran variedad de algoritmos de aprendizaje máquina, entre los cuales se encuentran los bosques aleatorios y el XGBoost, de los cuales se hablo en la introducción. Se utilizaron las librerias de *Python* de *sklearn* y *xgboost* para utilizar los modelos ya configurados por dichas librerías.

El modelo de bosques aleatorios utilizó 250 árboles en el bosque, la función de separación fue "Gini", el cual se calcula restando de uno la suma de las probabilidades al cuadrado de cada clase, por otra parte se configuró el

parámetro del número mínimo de muestras en cada separación de cada nodo interno a una cantidad igual a 100 y para reproducibilidad del modelo se utilizó un estado de aleatoriedad de 7.

El modelo de XGBoost se baso en un modelo basado en árboles, con un valor configurado de 250 estimadores y un estado de aleatoriedad de 7.

El interés principal de esta investigación es ver que algoritmo tiene mejores resultados con una configuración de parámetros similar.

4.6. Selección de Características Secuencial

Se empleó el algoritmo de Selección de Características Secuencial, en inglés se le conoce como *Sequential Feature Selection (SFS)*, pertenece a la familia de algoritmos de búsqueda voráz que se utilizan para reducir un espacio de características p-dimensional inicial a un subespacio de características k-dimensional donde $k < p$. Su objetivo principal es seleccionar el subconjunto de características que sea más relevante para el problema, lo que da como resultado una eficiencia de cálculo óptima y, al mismo tiempo, reduce el error de generalización al filtrar las características irrelevantes.

SFS se utilizó tanto para el algoritmo de bosque aleatorio como para el XGBoost, con el número máximo de 10 características, utilizando el método hacia delante y con la métrica de evaluación de exactitud, o más conocida en inglés como *accuracy*.

4.7. Back-Testing

La idea más sencilla del *back-testing* es entrenar al modelo con datos pasados hasta cierto momento y probarlo con datos desconocidos hasta ese momento. Se analizó un proceso de *back-testing* con ventana expandida de tal manera que se seleccionaron los primero 13 años para entrenar y un año para probar, después se utilizaron 14 años para entrenar y un año para probar y asi consecutivamente hasta llegar hasta el año 2021 para entrenar y el último año para probar. Se realizó dicha técnica para cada uno de los modelos de ML que se estudiaron en este artículo.

4.8. Diseño de experimentos

Se redistribuyeron los datos partiendo de la tabla 1 a la tabla 2 para llevar la técnica estadística adecuadamente.

Bosques Aleatorios	XG Boost	y_real
1	0	0
1	1	0
1	1	0
...	...	...

Cuadro 3: Datos iniciales

Variable	Valor
xgboost	0
bosque aleatorio	1
y_{real}	1
...	...

Cuadro 4: Datos transformados

Variable	Número de muestra	Media	Desviación estándar
y_real	2285	0.5448	0.4980
Bosque Aleatorio	2285	0.4582	0.4979
XG Boost	2285	0.4153	0.4928

Cuadro 5: Descriptivos por grupo.

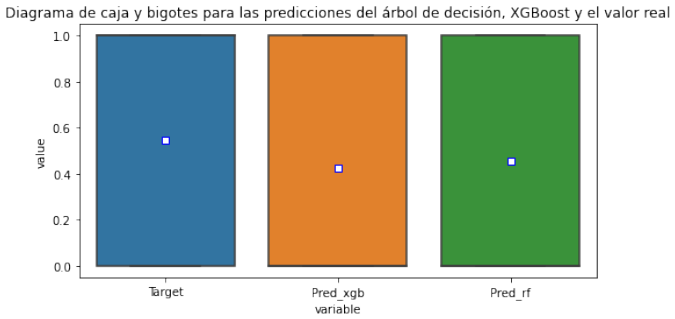

Figura 8: Gráfico de caja y bigotes para cada grupo

4.9. Descriptivos generales

Los datos en cada grupo se observan de la siguiente manera.

Es importante validar los supuestos de normalidad y homocedasticidad.

Para el supuesto de normalidad se utilizó la prueba de normalidad de Shapiro- Wilk y los resultados fueron que para bosques aleatorios, XG Boos y y_{real} los valores p son $2{,}17e-19$, $5{,}72e-18$ y $1{,}21e-14$ respectivamente, lo cual indica que se rechaza la hipótesis nula de que los valores provienen de una distribución normal.

Para la prueba de homocedasticidad presento un valor p de $0{,}65e-08$, lo cual indica que la varianza entre las grupos no es igual.

Dicha conclusiones ya se habian discutido al inicio del análisis sin embargo es importante llevar la metodología adecuadamente.

5. Resultados

Se utilizará la matriz de confusión como métrica principal para evaluar a los algoritmos de clasificación, la cual se describe en la figura 9. La matriz de confusión ayuda a visualizar y a resumir el rendimiento del algoritmo de clasificación. Otra forma de ver esta tabla es como la tabulación cruzada entre el valor real de la variable respuesta y la predicción.

Matriz de confusión			
		Reales	
		0	1
Predicción	0	Verdaderos Positivos (VP)	Falsos Positivos (FP)
	1	Falsos Negativos (FN)	Verdaderos Negativos (VN)

Figura 9: Matriz de confusión

Entre las métricas más importantes que es posible adquirir de la matriz de confusión es la exactitud, la cual se describe por la siguiente ecuación:

$$\text{Exactitud} = \frac{\text{Verdaderos positivos} + \text{Verdaderos negativos}}{\text{Total de muestras}}$$

La exactitud se describe el número correcto de predicciones.

5.1. Selección secuencial de características

Para el algoritmo de ML de bosques aleatorios se obtuvo que las características que más aportan al modelo son: máximo, día, diferencia de apertura y cierre, la proporción de cierre a 2,5,60,250 y 1000 días y la tendencia de 5 y 1000 días.

Por otro lado, para el algoritmo de XGBoost se obtuvo que las características que más aportaban eran: el cierre, mes, diferencia de apertura y cierre, la proporción de cierre a 2,5,60,250 y 1000 días y la tendencia de 250.

Se concluye que las características en común para ambos algoritmos que más aportan en la predicción del movimiento del precio de mañana son la diferencia de apertura y cierre, la proporción de cierre a 2,5,60,250 y 1000 días.

5.2. Back-testing para bosques aleatorios

La matriz de confusión resultante del proceso de *Back-testing* para bosques aleatorios con las características seleccionadas por SFS es la figura 10.

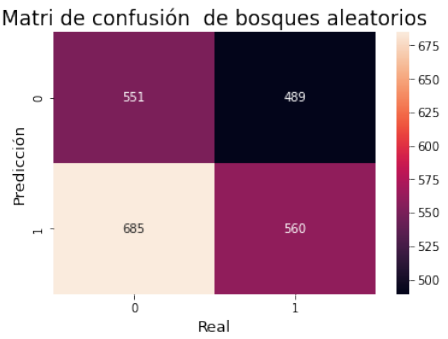

Figura 10: Matriz de confusión del algoritmo de Bosques aleatorios

La exactitud para este método resulto ser 48,32 %.

5.3. Back-testing para XGBoost

La matriz de confusión resultante del proceso de *Back-testing* para XGBoost con las características seleccionadas por SFS se observa en la figura 11.

El valor de exactitud para el método de XGBoost fue de 48,57 %.

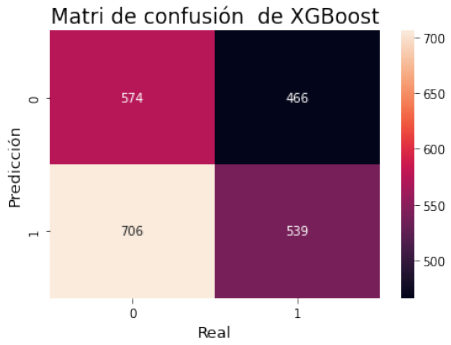

Figura 11: Matriz de confusión del algoritmo de XGBoost

5.4. Resultados ANOVA del diseño de experimentos.

Es importante destacar que los datos no siguen una distribución normal y no cumplen con el supuesto de homocedasticidad, además se conoce que se rechaza H_0 si pvalor es menor a α. En la tabla 9 se observa que el valor p es muy cercano a cero y el alfa de 0.05. Entonces, se rechaza H_0 , los grupos que se compararon no tienen la misma media.

Fuente	SS	DF	MS	F	p-unc	np2
variable	14.90	2	7.4533	49.05	4.9195e-19	0.2448
Within	45.13	297	0.1519			

Cuadro 6: Resultados del ANOVA

6. Conclusiones

Referencias

[1] M. L. Halls-Moore, Successfull Algorithmic Trading, 2020.
URL `https://www.quantstart.com/successful-algorithmic-trading-ebook/`

[2] J. Amar, Analisis de varianza anova con python, `https://www.cienciadedatos.net/documentos/pystats09-analisis-de-varianza-anova-` (2013).

[3] P. S. et al 2022", "machine learning approaches in stock price prediction a systematic review", 2022.

[4] Y.-S. L. Chin-Sheng H., Machine learning on stock price movement forecast: The sample of the taiwan stock exchange, International Journal of Economics and Financial 9 (2019) 189–201.

[5] e. a. Subasi A., Stock market prediction using machine learning, Procedia Compute Science ElSevier 194 (2021) 173–179.

[6] M. Kalirane, Ensemble learning methods: Bagging, boosting and stacking [cited 19032023].
URL https://www.analyticsvidhya.com/blog/2023/01/ensemble-learning-methods-bag

[7] A. Cutler, D. Cutler, J. Stevens, Random Forests, Vol. 45, 2011, pp. 157–176.

[8] S. Sameeruddin, How gradient boosting algorithm works [cited 20032023].
URL https://dataaspirant.com/gradient-boosting-algorithm/t-1606063032082