CMP5002 – Data Mining

Informe PSet #3

Integrantes: María Emilia Rivadeneira, Marie Cucalón, Roberth Lara, Melanie Alvarez

NRC: 2106

Fecha: 13/04/2025

URL GitHub: https://github.com/titusr099/project_trading_DT

URL Canva: https://www.canva.com/design/DAGkirtNmC8/SIW4Y8TmRKCl_k-YF5amXg/edit?utm_content=DAGkirtNmC8&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Contexto del Problema

Portafolio

Como parte del portafolio definido para este proyecto de trading basado en modelos de boosting, se seleccionaron cuatro activos con una participación equitativa del 25% cada uno: TLT (bonos del Tesoro a largo plazo), SPLV (ETF de baja volatilidad del S&P 500), JNJ (Johnson & Johnson), y Bitcoin.

TLT	SPLV	JNJ	Bitcoin
25%	25%	25%	25%

Definición del Target

Se definió un problema de clasificación binaria, donde el objetivo es predecir si el precio de cierre del siguiente día será mayor al del día actual. Esta formulación permite evaluar la capacidad de los modelos para anticipar subidas o bajadas en el precio del activo, lo cual es crítico para tomar decisiones de compra o venta en un entorno de trading automatizado.

Datos

Para cumplir con el objetivo propuesto, se obtuvieron datos de la biblioteca de yahoo finance, incorporada en python. Estos datos se remontan al 12 de diciembre de 2024, acumulando tres años de datos de entrenamiento y prueba hasta el presente; se eligió este periodo post-Covid de modo que las predicciones no se vieran alteradas por las fluctuaciones en el mercado durante la pandemia.

TLT (iShares 20+ Year Treasury Bond ETF)

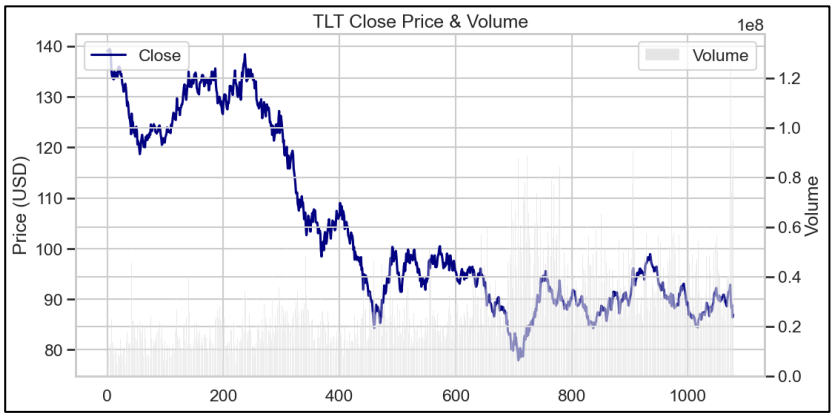
Durante casi un siglo, los T-Bonds han brindado un flujo de ingreso seguro y estable, sobre todo en periodos de incertidumbre económica. Los T-Bonds, o Bonos del Tesoro, son títulos de deuda de renta fija, emitidos por el gobierno federal de Estados Unidos a través del Departamento del Tesoro. Tienen un plazo de 20 o 30 años y pagan una tasa de interés cada seis meses hasta su vencimiento.

Básicamente, comprar un Bono del Tesoro es prestar dinero al gobierno, que lo utilizará para cubrir parte de sus gastos y a cambio pagará intereses, con el compromiso de devolver la totalidad del préstamo inicial al finalizar el plazo establecido.

1. ***Exploratory Data Analysis (EDA) y Data Wrangling***

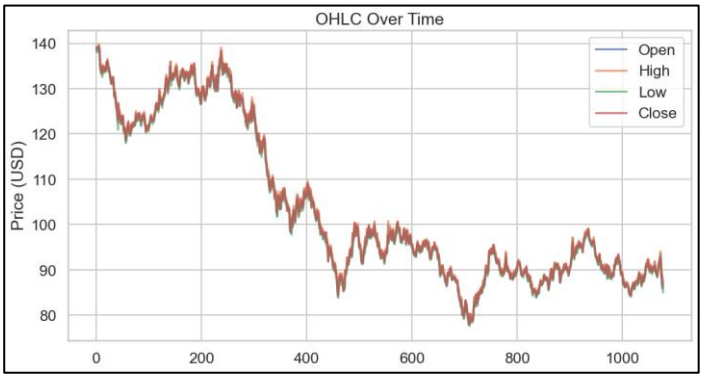
   A primera vista, se contaba con 1079 registros, cada uno con los campos 'Date', 'Close', 'High', 'Low', 'Open' y 'Volume'. Ninguna de las columnas mencionadas presentaba valores faltantes ni se encontrarion outliers.
   Se analizó la relación y comportamiento estadístico de estas variables:

   <u>Precio y volumen de cierre</u>



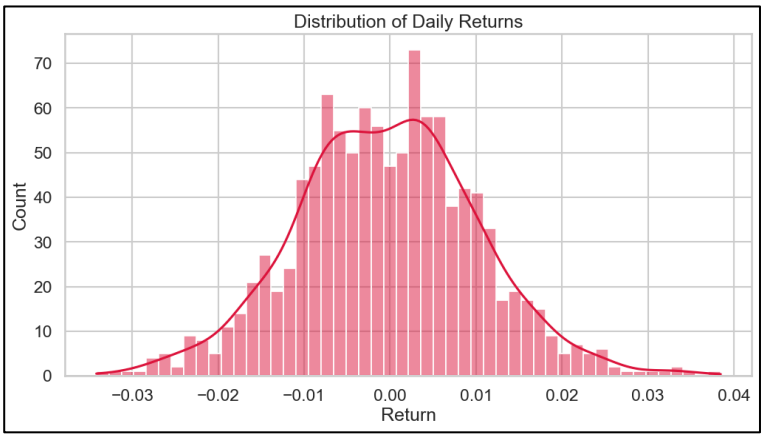
   El precio de TLT se comporta de forma inversa a las tasas de interés: sube cuando las tasas bajan, y cae cuando suben. El gráfico inicia con un precio alto y sufre una caída prolongada, lo que sugiere subida de tasas de interés o salida masiva de capital de bonos a largo plazo. El volumen aumenta considerablemente cuando cae el precio.

   <u>Open, Close, High, Low en el tiempo</u>



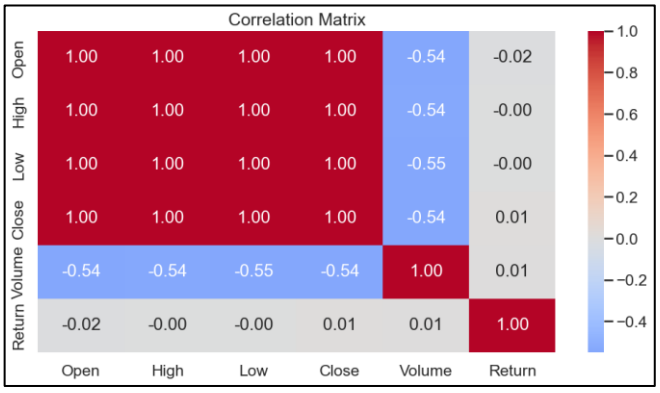
Este gráfico muestra el comportamiento diario del ETF. Las líneas están muy cercanas entre sí en la mayor parte del gráfico, es decir, la volatilidad intradía es relativamente baja, comportamiento característico de este bono del tesoro. El mercado parece estar consolidando en niveles bajos.

Histograma de retornos diarios



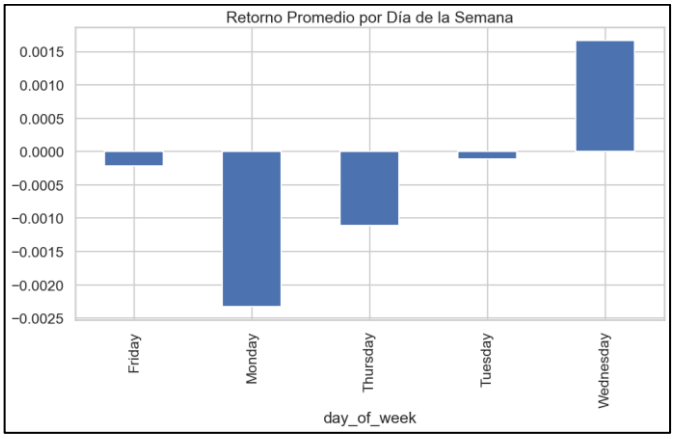
El gráfico muestra cómo varía el porcentaje de cambio en el precio de un día al otro. La distribución de retornos diarios muestra un comportamiento centrado en cero con ligera inclinación negativa, colas moderadas y evidencia de eventos extremos ocasionales. Aunque TLT es un activo relativamente estable, no está exento de días de alta volatilidad, especialmente en respuesta a cambios en tasas de interés.

Matriz de correlación



La matriz muestra que los precios diarios de TLT están fuertemente sincronizados entre sí, pero el volumen tiende a incrementarse cuando el precio baja. El retorno diario no guarda correlaciones lineales directas con precios o volumen, su predicción requiere modelos no lineales o el uso de indicadores técnicos o cambios porcentuales.

Retorno promedio por día de la semana



TLT empieza la semana con caídas considerables, se reduce considerablemente esta pérdida el martes, miércoles despuntan los retornos y vuelven a caer jueves y viernes. Esto muestra que el día en la semana influye en el precio de TLT.

2. *Feature Engineering*

Con el fin de aumentar la capacidad predictiva de los modelos, se aplicó ingenieria de características para convertir los datos crudos en información valiosa y representativa de las relaciones ocultas en los datos.

- Features técnicas: Derivadas matemáticas de series temporales.
  - **return_daily**: cambio porcentual diario del precio de cierre.
  - **return_lag_{n}**: retorno diario desplazado n días atrás.
  - **rolling_std_return_5**: volatilidad del retorno diario en una ventana de 5 días.
  - **return_volatility_ratio**: relación entre el retorno actual y su desviación estándar pasada.
  - **price_diff**: diferencia absoluta entre el precio de cierre y apertura.
  - **pct_diff**: diferencia relativa (porcentaje) entre cierre y apertura.
  - **log_vol**: volumen logarítmico (para estabilizar escalas y reducir outliers).
  - **sma_5**: media móvil simple del cierre en los últimos 5 días.
  - **ema_5**: media móvil exponencial del cierre en los últimos 5 días.
  - **rsi_5**: índice de fuerza relativa, mide momentum (sobrecompra/sobreventa) en 5 días.
  - **macd**: diferencia entre dos EMAs, usado para identificar tendencias.
  - **macd_signal**: línea de señal del MACD (EMA del MACD).
  - **macd_above_signal**: indica si el MACD está por encima de su señal (1) o no (0).
- Features temporales
  - **day_of_week:** día de la semana (0 = lunes, 4 = viernes).
  - **month:** mes del año.
  - **is_month_end:** indica si ese día es el final de mes (1) o no (0).

- Features macroeconómicas: Datos obtenidos con el API de Federal Reserve Economic Data.
    - **us10y_yield:** Tasa de interés de los bonos del Tesoro estadounidense a 10 años, que refleja las expectativas de crecimiento e inflación a largo plazo.
    - **fed_rate:** Tasa de referencia fijada por la Reserva Federal que influye en el costo del crédito y la política monetaria general.
    - **Cpi:** Índice de Precios al Consumidor que mide la inflación a través de la variación del precio de una canasta básica de bienes y servicios.
    - **unemployment_rate:** Porcentaje de personas desempleadas activamente buscando trabajo, que indica la salud del mercado laboral.

3. Modelling, Evaluation and Results

- **Carga y preparación de datos:** Se cargaron los datos limpios del ETF TLT, se dividieron en conjuntos de entrenamiento y prueba basados en la fecha, y se seleccionaron variables técnicas, macroeconómicas y temporales como features.
- **Validación:** Se utilizó TimeSeriesSplit para realizar validación cruzada respetando el orden temporal y se reservó el 10% final del set de entrenamiento como validación para early_stopping.
- **Optimización de hiperparámetros:** Cada modelo se ajustó mediante RandomizedSearchCV optimizando la precisión para la clase positiva (compra) usando validación temporal.
- **Entrenamiento con early_stopping:** Modelos como XGBoost, LightGBM y CatBoost usaron early_stopping sobre el set de validación para evitar sobreentrenamiento.
- **Threshold tuning:** Se probó un rango de umbrales (0.3 a 0.7) para encontrar el que maximizara la precisión macro en el test set.
- **Entrenamiento de múltiples modelos:** Se entrenaron y compararon 7 clasificadores: DecisionTree, RandomForest, AdaBoost, GradientBoosting, XGBoost, LightGBM, y CatBoost.
- **Almacenamiento de modelos:** Cada modelo entrenado fue guardado junto con su conjunto de features usando joblib en formato pkl.
- **Evaluación final:** Se generó un resumen comparativo mostrando métricas clave (precision_macro, recall_macro, f1_macro) y el mejor threshold para cada modelo.

Modelo	Threshold	Precisión Macro	Recall Macro	F1-score Macro
XGBoost	0.60	0.8269	0.6538	0.6306
GradientBoosting	0.30	0.8269	0.6538	0.6306
LightGBM	0.60	0.7931	0.5385	0.4410
RandomForest	0.70	0.7931	0.5385	0.4410
DecisionTree	0.55	0.7200	0.6244	0.6032
AdaBoost	0.45	0.5556	0.5204	0.4000
CatBoost	0.50	0.3891	0.3891	0.3891

4. Final Analysis

Se implementó una simulación financiera para evaluar cuánto dinero se habría ganado (o perdido) si se hubieran seguido las señales de compra generadas por cada modelo en función de su probabilidad de predicción (predict_proba) y distintos umbrales de decisión (thresholds).

📊 Resultados de la simulación con múltiples thresholds:

Modelo	Threshold	Trades realizados	Ganancia Total ($)	Capital Final ($)	Precision (1)
XGBoost	0.6	4	9.70	1009.70	1.000
RandomForest	0.6	5	6.14	1006.14	0.600
XGBoost	0.7	1	2.17	1002.17	1.000
GradientBoosting	0.7	1	2.17	1002.17	1.000
GradientBoosting	0.6	1	2.17	1002.17	1.000
GradientBoosting	0.5	1	2.17	1002.17	1.000
RandomForest	0.7	1	2.17	1002.17	1.000
DecisionTree	0.6	4	1.00	1001.00	0.750
DecisionTree	0.7	4	1.00	1001.00	0.750
RandomForest	0.5	10	0.69	1000.69	0.600
LightGBM	0.5	13	0.38	1000.38	0.615
AdaBoost	0.6	0	0.00	1000.00	0.000
AdaBoost	0.7	0	0.00	1000.00	0.000
CatBoost	0.7	0	0.00	1000.00	0.000
CatBoost	0.6	0	0.00	1000.00	0.000
LightGBM	0.7	0	0.00	1000.00	0.000
LightGBM	0.6	0	0.00	1000.00	0.000
XGBoost	0.5	13	-5.51	994.49	0.385
AdaBoost	0.5	12	-8.13	991.87	0.333
DecisionTree	0.5	23	-13.47	986.53	0.435
CatBoost	0.5	13	-13.59	986.41	0.308

Importante! Mayor precisión no significa mayor ganancia

La precisión mide cuántas predicciones fueron correctas en general, sin considerar la magnitud del retorno financiero de cada acierto. En trading, lo importante no es solo acertar si el precio subirá, sino cuánto sube cuando se acierta y cuánto se pierde cuando se falla. Un modelo puede tener alta precisión prediciendo movimientos pequeños o neutros, pero no captar oportunidades rentables grandes. Por eso, un modelo con menor precisión puede ser más lucrativo si sus aciertos ocurren en días con altos retornos positivos.

En este caso, coincidió que XGBoost resultó ser preciso y fructífero en inversión, pero no es una ley. La ganancia final fue de $9.70.

SPLV

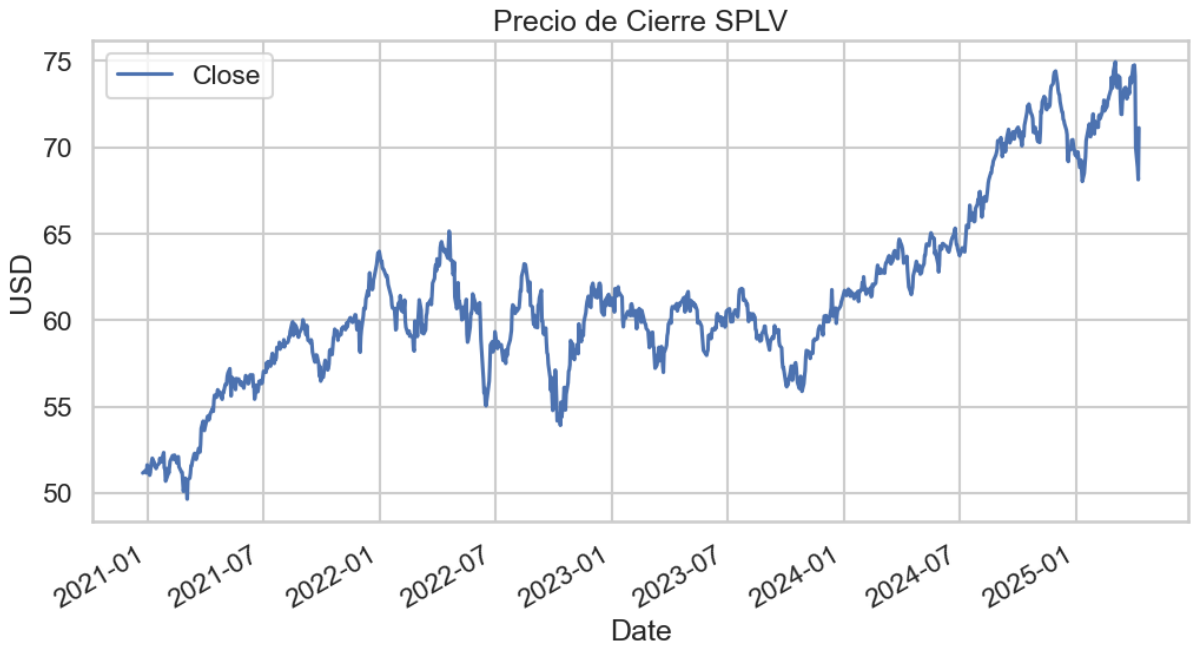
El ETF Invesco S&P 500® Low Volatility (SPLV) busca ofrecer una forma más estable de invertir en el mercado estadounidense. Este fondo replica el índice S&P 500® Low Volatility, que está formado por las 100 acciones del S&P 500 con menor volatilidad en los últimos 12 meses. En otras palabras, el SPLV prioriza acciones que han tenido menos altibajos en sus precios, y por eso se considera una opción más estable y menos riesgosa frente a otros fondos que siguen todo el S&P 500.
Este fondo se rebalancea cada tres meses (febrero, mayo, agosto y noviembre) para mantenerse actualizado, y suele estar compuesto por empresas de sectores como servicios públicos, consumo básico y salud, que normalmente se ven menos afectados por los cambios

en la economía. Gracias a esta estrategia, el SPLV tiende a tener movimientos más suaves en su precio, lo que lo hace ideal para personas o estrategias que buscan menor riesgo, especialmente en tiempos de incertidumbre.
Aunque esta menor volatilidad puede significar que no tenga grandes subidas cuando el mercado general está al alza, el SPLV es una buena opción para mantener cierta estabilidad en el portafolio. Es especialmente útil para estrategias conservadoras o de largo plazo, donde se quiere proteger el capital sin dejar de estar expuesto al crecimiento del mercado accionario.

1. EDA y Data Wrangling

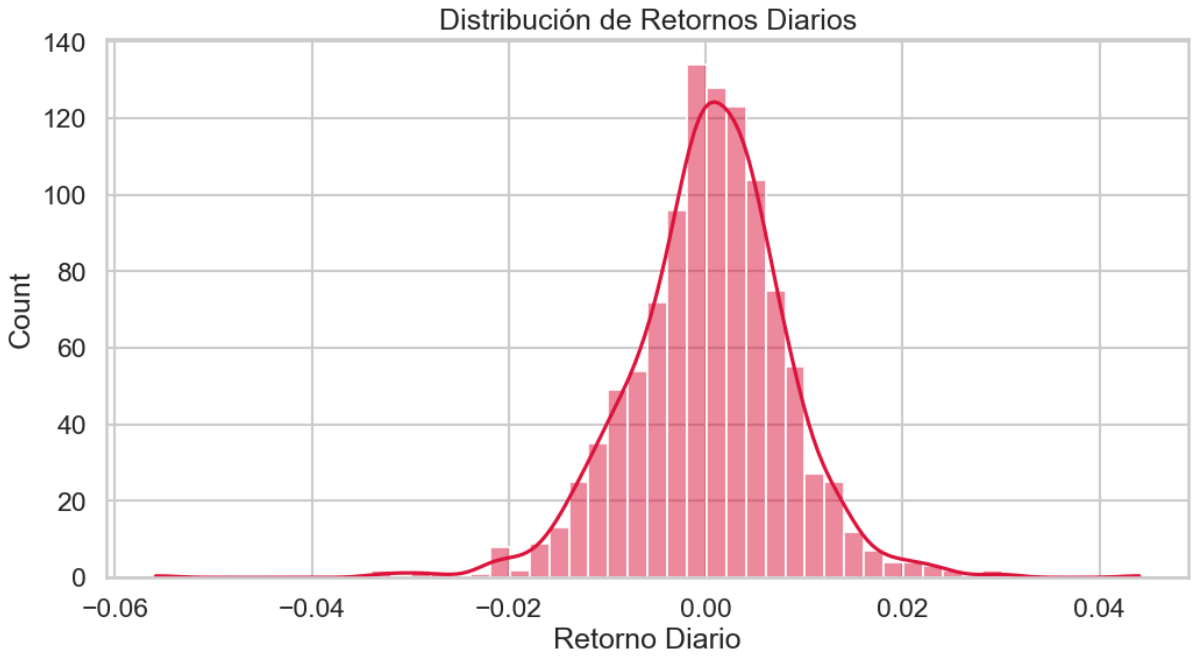
En este proyecto se ha realizado un análisis exploratorio de los datos históricos del ETF SPLV, obtenidos desde Yahoo Finance. El conjunto de datos cuenta con 1.077 registros diarios completos, sin valores faltantes, que abarcan desde el año 2021 hasta inicios de 2025. Las variables disponibles incluyen precios de apertura, cierre, máximos, mínimos y volumen negociado. El precio de cierre promedio es de 61.32 USD, con una desviación estándar de 5.32, lo que respalda la naturaleza de baja volatilidad del fondo. Los precios oscilaron entre un mínimo de 49.63 USD y un máximo de 74.95 USD durante el período analizado.
Aunque se detectaron varios valores atípicos en el volumen diario, no se aplicó una transformación logarítmica porque esta variable no se usará directamente en el modelo de predicción. En lugar de eso, se considera un dato adicional que puede ayudar a interpretar mejor ciertos movimientos del mercado. Además, como SPLV es un ETF de comportamiento estable, estos picos no distorsionan el análisis general ni afectan el rendimiento del modelo. Mantener el volumen en su forma original también facilita entender la magnitud real de los movimientos al momento de evaluar las estrategias.

A continuación, se verán las gráficas del análisis de los datos del SPLV:

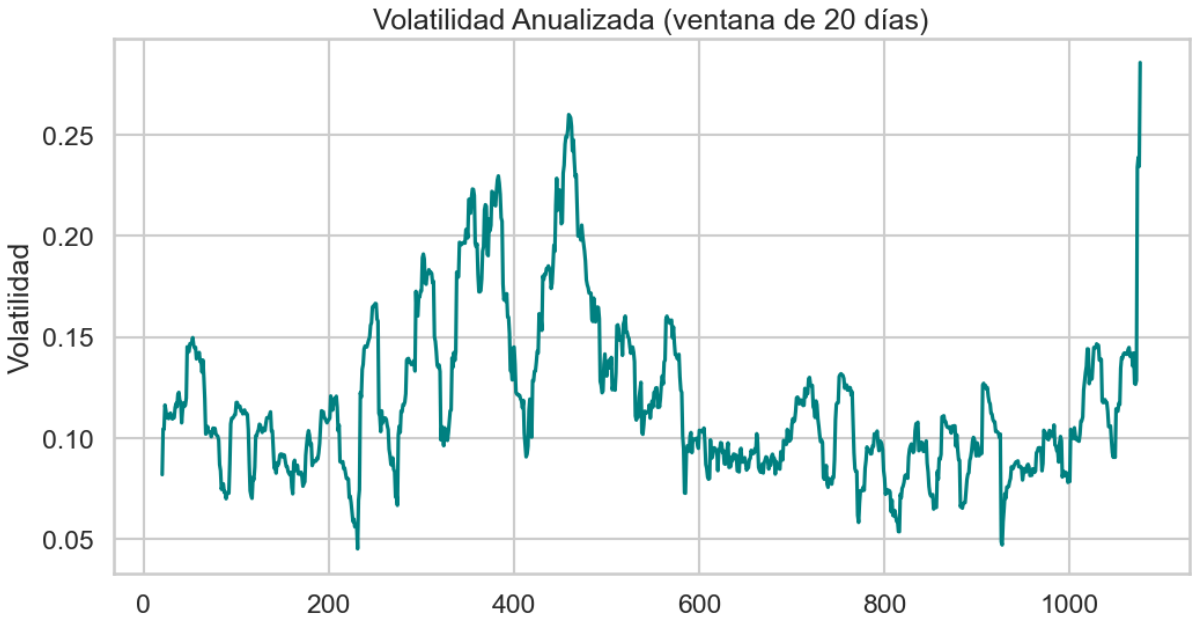


La gráfica muestra cómo ha cambiado el precio de cierre del ETF SPLV desde inicios de

2021 hasta principios de 2025. Se observa una tendencia general al alza, aunque con algunos momentos de caída. El precio pasó de alrededor de 50 USD a un máximo cercano a 75 USD. Esto confirma que, aunque SPLV es un fondo de baja volatilidad, sigue reaccionando a los cambios del mercado. Esta visualización fue útil para entender su comportamiento a largo plazo y definir el horizonte de predicción para el modelo.

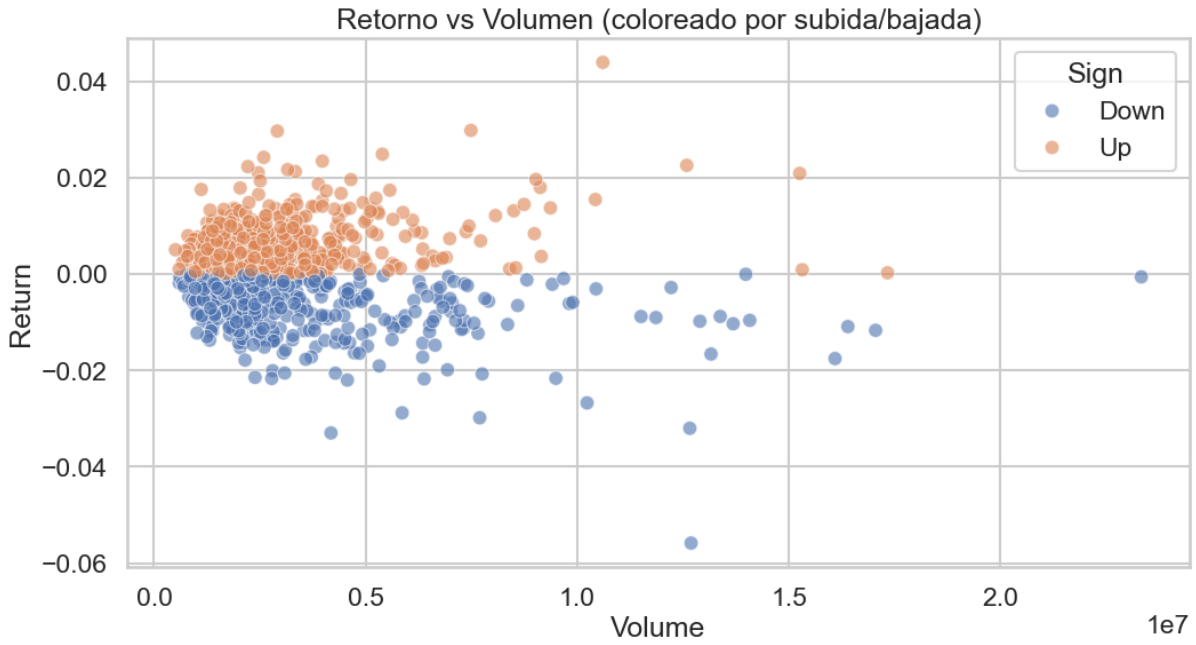


Este histograma muestra cómo se distribuyen los retornos diarios del ETF SPLV. La mayoría de los retornos diarios están entre -2% y +2%, con un pico alrededor del 0%. La forma tipo campana, centrada cerca de cero, indica que los movimientos diarios son en su mayoría pequeños, lo cual es consistente con un ETF de baja volatilidad. Esta información ayudó a confirmar que SPLV tiene un comportamiento estable y que es adecuado aplicar modelos de clasificación para predecir si el precio subirá o bajará.



Aquí se muestra cómo varió la volatilidad del ETF SPLV a lo largo del tiempo, utilizando una ventana móvil de 20 días. En general, la volatilidad se mantiene entre 0.08 y 0.18, con algunos picos que llegan hasta 0.26. Aunque SPLV presenta niveles bajos de riesgo, estos picos permiten identificar momentos más inestables. Esta información fue útil para considerar la inclusión de variables que capten esos cambios en el modelo.

Esta matriz muestra la relación entre las diferentes variables del ETF SPLV. Los precios de apertura, cierre, máximo y mínimo tienen una correlación muy alta entre sí (cercana a 1.00), lo que indica que se mueven casi igual y sugiere que no es necesario incluirlos todos en el modelo. En contraste, el volumen presenta una correlación baja con los precios (-0.20 a -0.22), y el retorno diario está casi desacoplado (~0.00 a 0.04), lo que indica que podrían aportar información adicional útil para la predicción.



En este gráfico se analizan los retornos diarios en función del volumen negociado, diferenciando los puntos por color según si el precio subió o bajó. Aunque no hay una separación clara, se observan ciertos patrones: los volúmenes muy altos tienden a estar asociados con días de baja, lo cual podría reflejar eventos externos o rebalanceos del fondo. Los retornos varían entre -6% y +4%, mientras que los volúmenes oscilan desde menos de 1 millón hasta más de 20 millones. Este análisis reforzó la decisión de mantener el volumen como variable sin transformarlo, ya que puede tener valor interpretativo en contextos específicos.

2. *Feature Engineering*

Para mejorar el rendimiento de los modelos de boosting aplicados al ETF SPLV, se diseñó un conjunto de variables que buscan capturar patrones de comportamiento relevantes en el contexto de un activo de baja volatilidad. Esta estrategia de *feature engineering* fue especialmente pensada para identificar cambios suaves pero consistentes en la tendencia, así como señales técnicas que anticipen posibles movimientos al alza o a la baja del precio de cierre.
El objetivo principal es predecir si el precio de cierre del día siguiente será mayor que el actual, lo cual sirve como señal para tomar decisiones de trading: comprar si se espera una subida o no operar si se espera una baja. Por ello, las variables generadas permiten al modelo aprender tanto del comportamiento reciente del mercado como de señales técnicas validadas en análisis bursátil.

Variables Temporales
Se incluyeron variables de calendario como **day_of_week, month** e **is_month_end**, con el objetivo de detectar patrones estacionales o comportamientos específicos asociados a ciertos días de la semana o al cierre de mes, donde pueden ocurrir rebalanceos del ETF o ajustes institucionales.

Diferencias de Precio y Retornos Históricos

Para capturar la dinámica reciente del precio, se calcularon variables como **price_diff, pct_diff, return_daily** y los rezagos **return_lag_1** hasta **return_lag_5.** Estas variables permiten al modelo observar la dirección y magnitud de los cambios recientes, lo cual es útil para detectar tendencias suaves o señales de reversión que suelen presentarse en SPLV.

Estadísticas Móviles e Indicadores Técnicos

Se generaron medias móviles como **sma_5** y **sma_10**, y la desviación estándar **rolling_std_return_5**, que ayudan a medir el momentum y la volatilidad reciente. Además, se incorporaron indicadores técnicos tradicionales:

- **RSI_5:** mide la fuerza del movimiento, útil para identificar condiciones de sobrecompra/sobreventa.
- **MACD y MACD_signal:** detectan cambios de tendencia.
- **Bollinger Bands:** representan rangos de precios esperados en base a la volatilidad.

Señales Técnicas Binarias

Se incluyeron variables binarias que resumen señales clave de trading como:

- **volume_outlier:** marca días con volúmenes anormales que podrían indicar eventos especiales.
- **price_above_SMA50**: identifica si el precio está en una tendencia alcista.
- **RSI_overbought y MACD_above_signal:** ayudan a detectar oportunidades de entrada o salida según condiciones técnicas.

Otras Variables Avanzadas

También se añadieron variables como **volatility_ratio, gap_up, lower_shadow, vol_change** y **sma_cross_up**, que buscan capturar señales sutiles de cambio de dirección, microtendencias y comportamiento de las velas japonesas. Aunque SPLV es un ETF de baja volatilidad, estos patrones de vela pueden seguir apareciendo en su comportamiento diario, aunque de forma más sutil.

3. Modelling

En esta sección se implementaron diversos modelos de clasificación para predecir si el precio del ETF SPLV subirá o bajará al día siguiente. El enfoque del proyecto se centró en maximizar la precisión de las señales positivas (clase 1), es decir, reducir los falsos positivos, para tomar decisiones de compra más confiables dentro de una estrategia de trading conservadora.

Preparación y División de Datos

Los datos procesados se dividieron temporalmente en dos subconjuntos:

- **Conjunto de entrenamiento:** incluye los datos hasta el 28 de febrero de 2025.
- **Conjunto de prueba:** cubre desde el 1 de marzo de 2025 en adelante.

Se eliminaron variables como Close, Open, High, Low y Date del conjunto de características, ya que contienen información contemporánea al objetivo y no deben usarse para predecir el valor futuro. El objetivo (target) es binario: **1 si el precio sube al día siguiente, 0 si baja o se mantiene.**

Modelos Utilizados

Se construyó un diccionario de modelos que incluyó:

- Árbol de Decisión
- Random Forest
- AdaBoost
- Gradient Boosting
- LightGBM
- XGBoost

Cada modelo fue configurado con hiperparámetros adecuados para evitar sobreajuste y mejorar su capacidad de generalización, incluyendo:

- **class_weight='balanced'** en DecisionTree, RandomForest y LightGBM para tratar el desbalance de clases.
- **scale_pos_weight** en XGBoost, calculado como la razón entre clases (mayoría/minoría) y **suavizado con un factor de 0.5** para evitar penalizaciones excesivas sobre la clase negativa.
- **eval_metric='aucpr'** en XGBoost, para enfocarse en la detección de la clase positiva en un entorno desbalanceado.
- **Búsqueda aleatoria de hiperparámetros (RandomizedSearchCV)** aplicada a LightGBM, usando f1_macro como métrica de optimización, con validación cruzada de 3 folds.

Evaluación con Threshold Ajustado

Con el objetivo de reducir los falsos positivos (compras innecesarias), se modificó el threshold de clasificación a **0.55** (en lugar del valor estándar de 0.5). Esto significa que una instancia solo se clasifica como clase 1 (subida) si la probabilidad predicha supera el 55%. La función train_and_evaluate() fue adaptada para aceptar este threshold y calcular las siguientes métricas para cada modelo:

- Accuracy
- Precisión (macro)
- Recall (macro)
- F1 Score (macro)
- Matriz de Confusión
- Reporte de Clasificación
- Importancia de Variables (cuando está disponible)

Los modelos entrenados fueron guardados en formato .pkl en la carpeta models/ para su posterior validación o reutilización.

Mejores Modelos

Con base en los resultados obtenidos usando un threshold de 0.55, los dos **modelos con mejor desempeño según la precisión macro (precision_macro)** fueron:

- Gradient Boosting
- XGBoost

Estos modelos demostraron una buena capacidad para detectar correctamente los días en que el precio del SPLV sube, sin comprometer excesivamente el recall.

En particular, el modelo **Gradient Boosting** fue el que alcanzó la mejor precisión macro, con un valor de **0.8269**, siendo el más confiable en decisiones de compra dentro del contexto de este proyecto de trading algorítmico basado en boosting.

4. Evaluation and Results

Tras entrenar todos los modelos con un umbral de clasificación ajustado a 0.55, se evaluó el desempeño de cada uno sobre el conjunto de prueba usando métricas estándar: accuracy, precision, recall, y f1-score en su versión macro. Esta elección permite evaluar el rendimiento balanceado entre ambas clases, especialmente útil cuando se desea evitar falsos positivos en decisiones de compra.

A continuación, se resumen los resultados más importantes:

- **GradientBoosting** y **XGBoost** obtuvieron la mayor accuracy (0.6786), destacándose también en precisión, con valores de 0.8269 y 0.7083 respectivamente.
- **DecisionTree** y **RandomForest** mostraron un equilibrio entre precisión y recall, ambos con f1-score macro de 0.6257.
- **LightGBM** fue el modelo con menor desempeño general, a pesar de tener buen recall (0.5856), su precisión fue más baja.

Matrices de Confusión

Las matrices de confusión revelan cómo se distribuyen los errores entre clases. Por ejemplo:

- **GradientBoosting** acertó todos los casos negativos (clase 0) pero falló en 9 de 11 casos positivos.
- **LightGBM** cometió más errores totales (13), especialmente clasificando mal ejemplos de clase 0.
- **RandomForest** logró 4 verdaderos positivos, pero tuvo 7 falsos negativos.
- **XGBoost** presentó un comportamiento intermedio: clasificó correctamente 14 de los 17 casos negativos y 4 de los 11 positivos. Cometió 3 falsos positivos y 7 falsos negativos, lo cual indica una preferencia por minimizar compras erróneas, aunque a costa de dejar pasar oportunidades reales de subida.

Estas observaciones permiten identificar qué modelos son más conservadores (alta precisión, bajo recall) o más arriesgados.

Curvas ROC

Las curvas ROC indican la capacidad de los modelos para distinguir entre clases. Los mejores AUC fueron:

- **XGBoost**: AUC = 0.64
- **RandomForest / DecisionTree**: AUC = 0.63
- **GradientBoosting**: AUC = 0.59

Optimización del Threshold

Se graficó la curva **Precision vs Recall** para los modelos más prometedores. En general:

- A medida que se incrementa el threshold, la precisión mejora pero el recall disminuye.
- **XGBoost** alcanza su mejor tradeoff entre 0.5 y 0.6.
- **GradientBoosting** pierde recall rápidamente al subir el umbral.

Errores de Clasificación

Se registraron errores de predicción por modelo. **GradientBoosting** tuvo el menor número de errores (9/28), seguido por **DecisionTree**, **RandomForest** y **XGBoost** (todos con 10). Esto valida su capacidad para reducir falsos positivos y mejora la fiabilidad para estrategias de inversión.

5. ***Final Analysis***

En esta sección se llevó a cabo la simulación de una estrategia de trading basada en las predicciones de los modelos de clasificación entrenados previamente. El objetivo fue evaluar el rendimiento financiero de cada modelo al invertir un 25% de un capital inicial de $1000 en días en los que el modelo predice una subida del precio de cierre (clase 1).

Estrategia de Simulación

- **Capital inicial:** $1000
- **Fracción de inversión por operación:** 25%
- **Ejecución de compra:** solo si la predicción fue positiva (clase 1)
- **Ganancia/Pérdida:** calculada con el cambio porcentual entre el precio de cierre del día actual y el siguiente

Se utilizaron las probabilidades (**predict_proba**) de cada modelo con un umbral de decisión ajustado (threshold = 0.55), con el objetivo de minimizar falsos positivos y asegurar mayor precisión en las compras.

Resultados Globales por Modelo

Modelo	Trades	Wins	Losses	Profit ($)	Capital Final ($)

RandomForest	**6**	**6**	**2**	15.75	1015.75
XGBoost	**4**	**4**	**1**	14.78	1014.78
GradientBoosting	**2**	**2**	**0**	3.27	1003.27
AdaBoost	**0**	**0**	**0**	0.00	1000.00
LightGBM	**14**	**6**	**8**	-10.12	989.88
DecisionTree	**11**	**6**	**4**	-12.45	987.55

Los modelos más rentables fueron **RandomForest** y **XGBoost**, con ganancias netas superiores a $14. El modelo **GradientBoosting** también fue rentable, pero con menos operaciones. En contraste, **LightGBM** y **DecisionTree** mostraron pérdidas, pese a generar más señales de compra.

Análisis de Thresholds

Se exploraron múltiples umbrales entre 0.35 y 0.60 para los modelos más prometedores (XGBoost, GradientBoosting y RandomForest) con el fin de encontrar el equilibrio óptimo entre precisión, recall y rendimiento financiero.

📈 Comparación de thresholds en XGBoost:

Threshold	Accuracy	Precision	Recall	F1-score	Trades	Wins	Losses	Profit ($)	Final Capital
0.55	0.6786	0.7500	0.2727	0.4000	4	3	1	14.78	1014.78
0.60	0.6786	0.7500	0.2727	0.4000	4	3	1	14.78	1014.78
0.50	0.6429	0.5714	0.3636	0.4444	7	4	3	8.57	1008.57
0.40	0.5357	0.4444	0.7273	0.5517	18	8	9	-3.68	996.32
0.45	0.5714	0.4615	0.5455	0.5000	13	6	7	-5.32	994.68
0.35	0.4643	0.4000	0.7273	0.5161	20	8	11	-7.00	993.00

📈 Comparación de thresholds en GradientBoosting:

Threshold	Accuracy	Precision	Recall	F1-score	Trades	Wins	Losses	Profit ($)	Final Capital
0.50	0.6786	1.0000	0.1818	0.3077	2	2	0	3.27	1003.27
0.55	0.6786	1.0000	0.1818	0.3077	2	2	0	3.27	1003.27
0.60	0.6786	1.0000	0.1818	0.3077	2	2	0	3.27	1003.27
0.45	0.6429	0.6667	0.1818	0.2857	3	2	1	2.56	1002.56
0.40	0.5714	0.4000	0.1818	0.2500	5	2	3	-4.55	995.45
0.35	0.5000	0.3333	0.2727	0.3000	9	3	6	-23.64	976.36

📈 Comparación de thresholds en RandomForest:

Threshold	Accuracy	Precision	Recall	F1-score	Trades	Wins	Losses	Profit ($)	Final Capital
0.55	0.6786	0.6667	0.3636	0.4706	6	4	2	15.75	1015.75
0.50	0.6429	0.5455	0.5455	0.5455	11	6	4	14.69	1014.69
0.45	0.5714	0.4706	0.7273	0.5714	17	8	8	11.69	1011.69
0.40	0.5000	0.4286	0.8182	0.5625	21	9	11	5.99	1005.99
0.60	0.6429	0.6667	0.1818	0.2857	3	2	1	3.67	1003.67
0.35	0.4286	0.4000	0.9091	0.5556	25	10	14	-8.83	991.17

- XGBoost alcanzó su mejor rentabilidad con un threshold de 0.55–0.60.
- Con un threshold conservador (0.55), GradientBoosting fue preciso, pero con bajo recall. Esto resultó en menor número de operaciones, pero sin pérdidas.
- RandomForest fue el modelo más consistente, generando mayores beneficios con distintos umbrales. El umbral óptimo se encontró en 0.55.

Para un enfoque conservador basado en precisión, bajo riesgo y rentabilidad positiva, el modelo óptimo para predecir subidas en el ETF SPLV es XGBoost.

Conclusiones

- El ETF SPLV tiene pocos cambios bruscos, así que es perfecto para invertir con poco riesgo y buscar estabilidad en el tiempo.
- Con un umbral ajustado, XGBoost dio pocas señales, pero muy acertadas. Generó ganancias sin asumir riesgos innecesarios. Aunque es menos preciso que XGBoost, da más oportunidades de compra y fue el que más dinero generó en la simulación.
- Modelos como LightGBM y DecisionTree hicieron muchas predicciones fallidas. No son viables para esta estrategia.
- Es preferible tener menos compras, pero seguras. Así se evitan pérdidas y se mantiene el capital protegido.

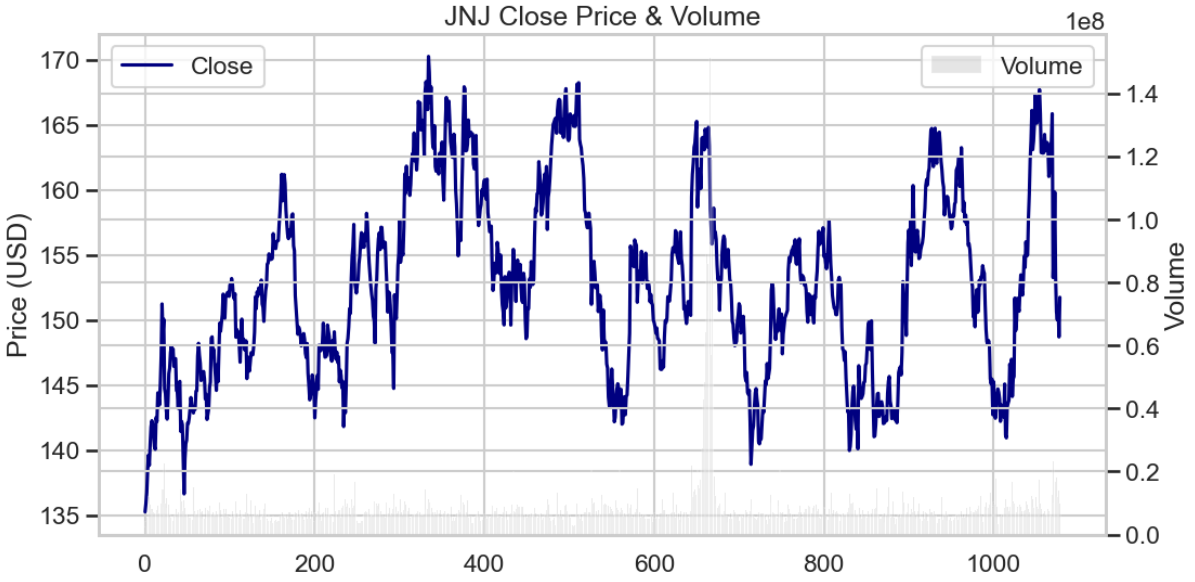
JNJ

Exploratory Data Analysis (EDA) y Data Wrangling

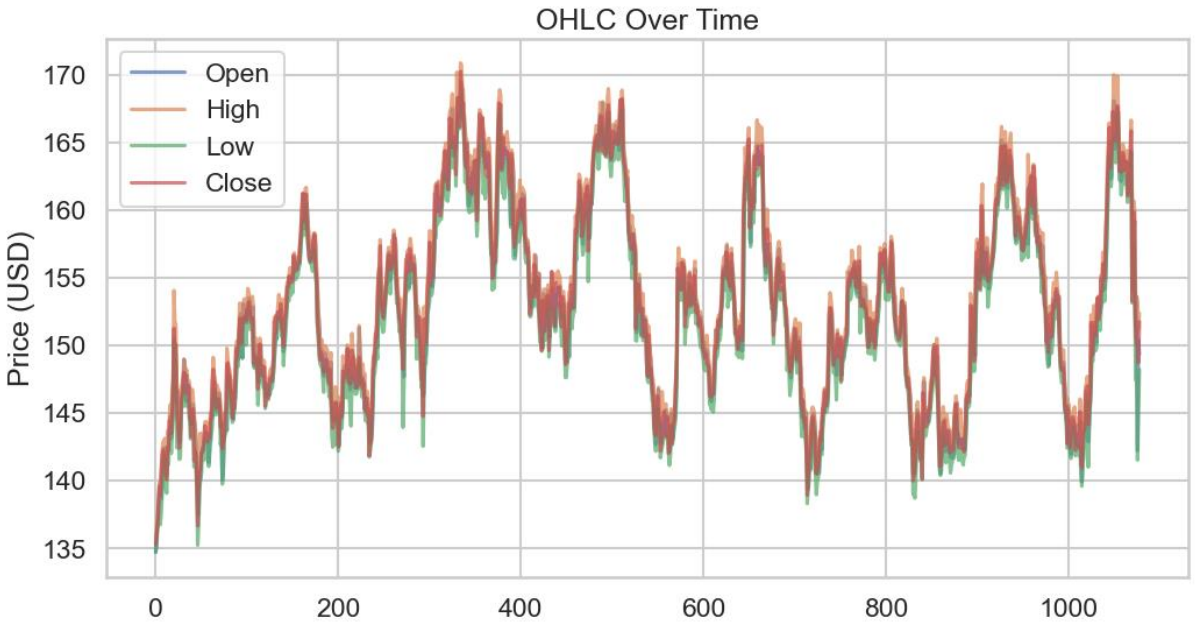
Los datos históricos de JNJ se obtuvieron de Yahoo Finance e incluyen 1,078 registros diarios, sin presencia de valores nulos, lo cual permite un análisis continuo sin necesidad de imputación. Además, al realizar un análisis de las variables disponibles por día, que incluyen: Date, Open, High, Low, Close y Volume; se evidencia que esta última variable es la única que presenta outliers. Por lo tanto ha decidido que no se eliminarán, pues son precisamente los movimientos del mercado que el modelo debe detectar. En su lugar, se controlarán al suavizar sus valores mediante el uso de logaritmos.

Por otro lado, el análisis estadístico preliminar revela que el precio de cierre promedio ha sido de aproximadamente 152.85 USD, con una desviación estándar de 7.08 USD, lo cual indica una volatilidad moderada. Además, el volumen de negociación muestra una dispersión significativa, con valores que van desde 2.1 millones hasta más de 151 millones, lo cual podría reflejar eventos corporativos o macroeconómicos relevantes.

Algunos de los gráficos más importantes junto con sus respectivos análisis son los siguientes:

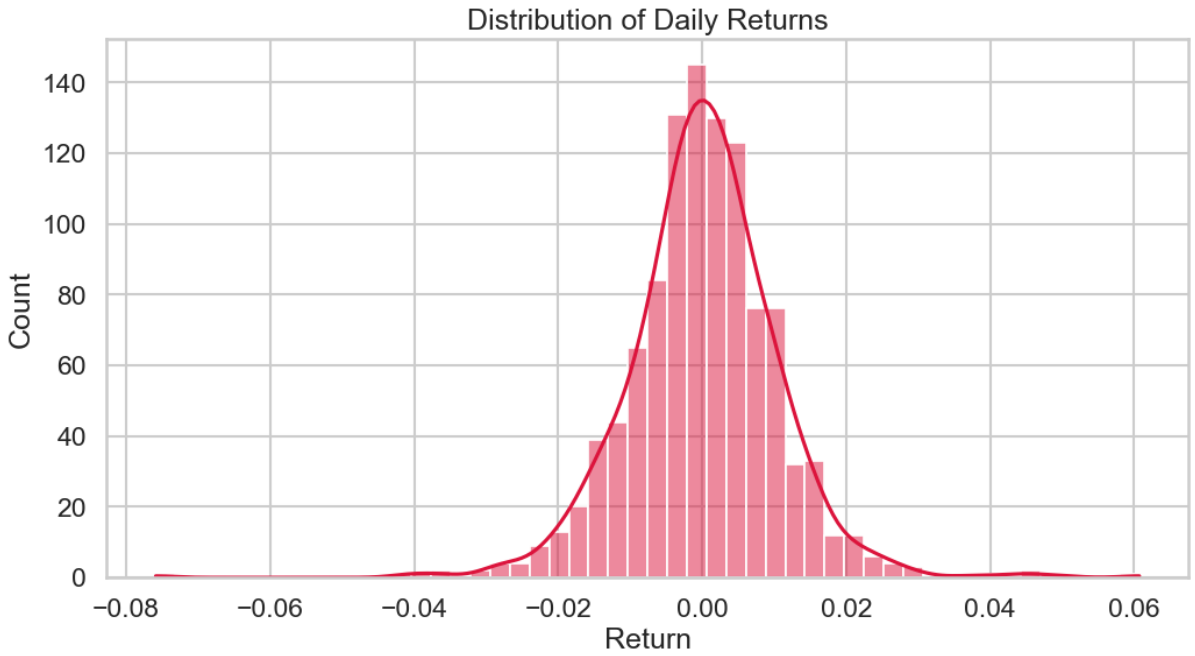


En este gráfico de precio de cierre y volumen de JNJ se aprecia que, a lo largo del período analizado, el precio oscila dentro de un canal aproximado de 135 – 170 USD, con varios ciclos de alzas sostenidas seguidas de correcciones pronunciadas. Los picos de volumen (barras más altas) tienden a coincidir con los puntos de inflexión del precio, indicando que los días de mayor negociación suelen marcar inicios o cierres de tendencias. En contraste, durante los tramos laterales o de consolidación el volumen se mantiene relativamente bajo, lo que sugiere menor interés de mercado y menor volatilidad en el precio. Este comportamiento es típico en acciones de gran capitalización: movimientos impulsados por noticias o resultados trimestrales generan volumen elevado y cambios bruscos de precio, mientras que en períodos neutrales el mercado se estabiliza.

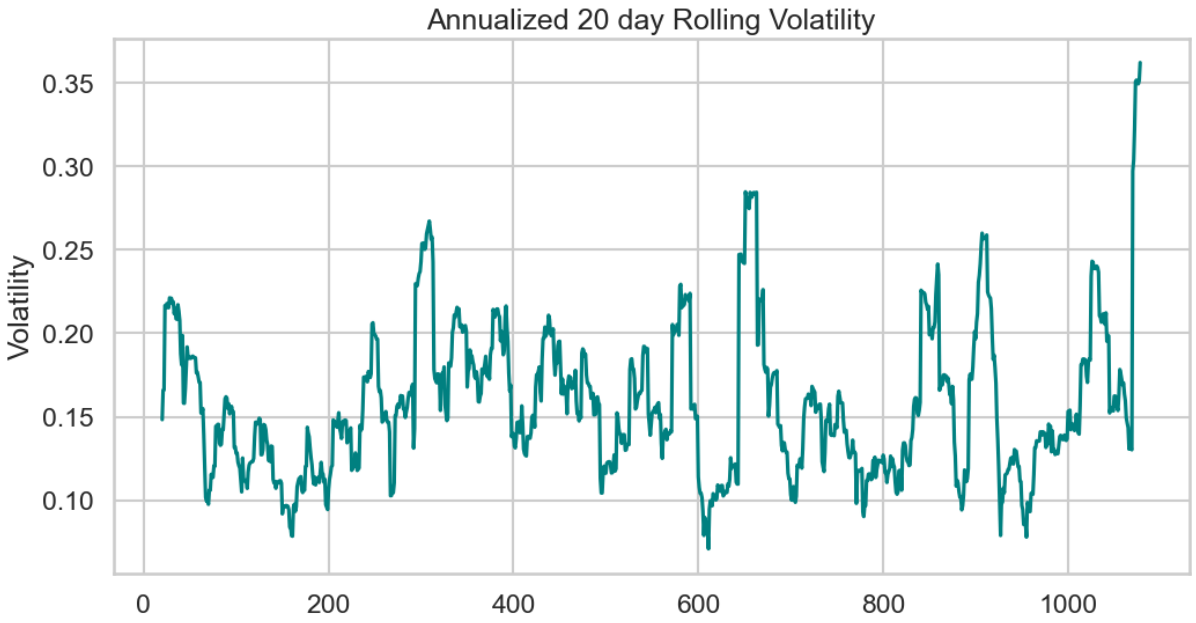


En la superposición de las series Open, High, Low y Close se aprecia que los cuatro precios se mueven casi solapados, lo que indica que los “gaps” intradía (diferencias entre cierre y apertura siguiente) son poco frecuentes y de tamaño reducido. El rango diario (High–Low)

también es relativamente estrecho, señal de baja volatilidad intradía en la mayoría de las sesiones. Sin embargo, a lo largo del período se distinguen claros ciclos de tendencia alcista y bajista: cuando el precio sube o baja con fuerza, las cuatro líneas se separan un poco más, mostrando velas con cuerpos y mechas más largos. En los tramos laterales, en cambio, todas convergen y el mercado entra en consolidación.



La distribución de retornos diarios de JNJ está muy concentrada alrededor de cero, con la mayoría de los días mostrando variaciones pequeñas (±1 %). Sin embargo, se aprecian colas relativamente gruesas: hay más ocurrencias de movimientos extremos (tanto alcistas como bajistas) de las que cabría esperar en una distribución normal. Además, la leve asimetría hacia la izquierda (cola negativa algo más pronunciada) sugiere que las caídas fuertes son un poco más frecuentes o bruscas que las subidas equivalentes. En conjunto, esto indica un comportamiento leptocúrtico y un ligero sesgo negativo, factores importantes a considerar al modelar riesgos y diseñar estrategias de trading.



En la serie de volatilidad anualizada a 20 días se observan claros ciclos de riesgo: períodos de baja volatilidad (por debajo de 0.12–0.15) alternan con picos pronunciados que superan 0.20 e incluso 0.30, reflejando momentos de alta incertidumbre o noticias relevantes para JNJ. Los valles prolongados sugieren fases de mercado tranquilo y consolidación, donde los retornos diarios son más predecibles, mientras que los máximos abruptos marcan episodios de tensión —quizá en torno a resultados trimestrales o eventos macro— que podrían desencadenar estrategias de protección o rotación de posiciones. Este comportamiento no estacionario indica que cualquier modelo predictivo debe adaptarse dinámicamente a cambios en la volatilidad para no subestimar el riesgo en los picos.

En la matriz de correlación se ve que las cuatro cotizaciones diarias (Open, High, Low y Close) están casi perfectamente alineadas (correlaciones entre 0.98 y 0.99), lo cual refleja que

cualquier cambio en el precio afecta de modo muy parecido a todos esos valores. El volumen muestra una correlación débilmente positiva (0.10) con los precios, indicando que los días de mayor negociación suelen coincidir con ligeros movimientos de precio, pero no de forma determinante. Por su parte, los retornos diarios están prácticamente no correlacionados con los niveles de precio (cerca de 0.00) y tienen una correlación levemente negativa con el volumen (–0.08), sugiriendo que las jornadas de alta actividad tienden a asociarse con caídas algo más frecuentes que con subidas.

Feature Engineering

Con el objetivo de mejorar la capacidad predictiva del modelo de boosting, se llevó a cabo un proceso de feature engineering que permite capturar patrones de comportamiento en el precio de la acción de Johnson & Johnson (JNJ). Las variables generadas abarcan variables temporales, diferencias de precio, indicadores técnicos y banderas binarias que reflejan condiciones específicas del mercado.

Variables Temporales

En primer lugar, se incorporaron variables de calendario, como por ejemplo: day_of_week, month, is_month_end; para capturar posibles patrones estacionales, efectos del día de la semana y comportamientos asociados al fin de mes que podrían influir en el movimiento del precio.

Diferencias de Precio y Retornos Históricos

Se calcularon diferencias absolutas (price_diff) y relativas (pct_diff) entre los precios de apertura y cierre, así como los retornos diarios (return_daily) y sus rezagos (return_lag_1 hasta return_lag_5). Estas variables permiten al modelo aprender dinámicas recientes en los cambios porcentuales del precio, lo que es clave para detectar tendencias o reversión a la media en el corto plazo.

Estadísticas Móviles e Indicadores Técnicos

Se incluyeron promedios móviles (sma_5) y desviaciones estándar (rolling_std_return_5) que resumen el comportamiento de las últimas jornadas bursátiles, útiles para detectar volatilidad y momentum. Asimismo, se incorporaron indicadores técnicos ampliamente utilizados en análisis técnico:

- RSI (Relative Strength Index): mide la fuerza del movimiento de precios, ayudando a identificar condiciones de sobrecompra o sobreventa.
- MACD (Moving Average Convergence Divergence) y su señal: capturan cambios en la fuerza, dirección y duración de una tendencia.

- Bollinger Bands: delimitan rangos de precios en función de la volatilidad reciente, útiles para detectar rupturas o rebotes.

Banderas Binarias: Señales de Trading

Se construyeron variables binarias que representan señales relevantes para decisiones de trading:

- volume_spike: identifica días con volumen de negociación inusualmente alto, lo cual puede indicar interés institucional o eventos relevantes.
- price_above_SMA50: señala si el precio se encuentra por encima de su media móvil de 50 días, asociado a una posible tendencia alcista.
- RSI_overbought: indica si el RSI supera el umbral de 70, condición comúnmente relacionada con sobrecompra.
- MACD_above_signal: marca cuando el MACD cruza por encima de su línea de señal, lo cual puede interpretarse como una señal de compra.

Modelling

Para garantizar una evaluación justa y libre de sesgos temporales, los datos fueron divididos en un conjunto de entrenamiento que contiene datos hasta febrero de 2025, y un conjunto de prueba correspondiente a marzo de 2025. Esta separación temporal simula un escenario de inversión real donde las predicciones deben realizarse sin conocer datos futuros.

En particular, se entrenaron y evaluaron siete modelos distintos, y para maximizar el rendimiento de cada modelo y adaptarlo a la naturaleza específica de los datos financieros de JNJ, se definieron espacios de búsqueda personalizados para sus hiperparámetros clave. Estos espacios fueron evaluados utilizando búsqueda aleatoria (RandomizedSearchCV) combinada con validación cruzada estratificada, asegurando una optimización robusta sin sobreajuste. A continuación se describen a detalle:

Árbol de Decisión (Decision Tree)

- criterion: define la función de impureza para dividir nodos. Se prueban tres alternativas: gini (rápido y común), entropy (basado en información), y log_loss (relacionado con la probabilidad).
- max_depth: limita la profundidad del árbol, lo cual regula la complejidad. Se evalúan profundidades pequeñas (3, 5), medianas (10, 20), y sin límite (None).
- min_samples_split y min_samples_leaf: controlan el tamaño mínimo para realizar divisiones y formar hojas, respectivamente. Esto evita ramas muy pequeñas que tienden al sobreajuste.
- max_features: determina cuántas variables considerar al dividir, introduciendo aleatoriedad (sqrt, log2) o considerando todas (None).

- class_weight: permite balancear las clases automáticamente, útil en casos de desbalance en las subidas y bajadas.

Random Forest

- n_estimators: número de árboles (200–800), donde más árboles tienden a mejorar la estabilidad del modelo.
- max_depth, min_samples_split, min_samples_leaf: mismos conceptos que en Decision Tree, aplicados para controlar el crecimiento individual de cada árbol.
- max_features: define qué fracción de variables usar en cada split.
- bootstrap: determina si se realiza muestreo con reemplazo (clásico de Random Forest).
- class_weight: al igual que antes, ajusta por clases desbalanceadas.

AdaBoost

- n_estimators: número de clasificadores secuenciales.
- learning_rate: controla cuánto peso tiene cada clasificador (una tasa más baja suaviza el aprendizaje).
- estimator__max_depth, estimator__min_samples_split, estimator__min_samples_leaf: controlan la complejidad del clasificador base (en este caso, árboles simples).
- estimator__class_weight: añade manejo de desbalance al clasificador base.

Gradient Boosting

- n_estimators y learning_rate: definen cuántos árboles se generan y qué tan fuerte contribuye cada uno.
- max_depth, min_samples_split, min_samples_leaf: previenen overfitting al regular la complejidad de cada árbol.
- subsample: al usar valores menores a 1.0, se introduce aleatoriedad tipo *stochastic boosting*, que mejora la generalización.
- max_features: regula la diversidad entre árboles.

LightGBM

- num_leaves: determina la cantidad de hojas en cada árbol, afectando la complejidad.
- min_child_samples: especifica el número mínimo de registros por hoja, útil para regularización.
- feature_fraction y bagging_fraction: indican qué fracción de características y observaciones usar, respectivamente, en cada iteración.
- bagging_freq: define cada cuántas iteraciones aplicar bagging.

- lambda_l1 y lambda_l2: coeficientes de regularización L1 y L2.
- min_gain_to_split: controla el umbral mínimo de ganancia requerido para realizar una división.
- max_depth, n_estimators, learning_rate: igual que en otros métodos de boosting.

XGBoost

- n_estimators, learning_rate, max_depth: clásicos en boosting.
- subsample, colsample_bytree: permiten muestreo aleatorio por filas y columnas, ideal para reducir sobreajuste.
- scale_pos_weight: ajusta el peso de la clase minoritaria; se evaluaron valores 1 y 5 para controlar posibles desbalances.

CatBoost

- iterations y learning_rate: control del número y fuerza de árboles.
- depth: profundidad del árbol de decisión.
- l2_leaf_reg: regularización L2 aplicada a las hojas del árbol.
- bagging_temperature: regula la aleatoriedad del bootstrap.
- random_strength: controla cuánto influye el *random noise* al seleccionar variables.
- border_count: cantidad de divisiones (binarización) para variables continuas.
- grow_policy: define cómo se estructura el árbol (simbólico, por profundidad, o basado en pérdida).
- eval_metric: define si se prioriza exactitud (Accuracy) o equilibrio en clases (F1).

Evaluation and Results

Tras entrenar todos los modelos de clasificación con sus respectivos hiperparámetros ajustados, se procedió a evaluar su desempeño sobre el conjunto de prueba (marzo de 2025).

Los resultados, junto con su respectivo análisis son los siguientes:

Modelo	Accuracy	Precision Macro	Recall Macro	F1 Macro
CatBoost	0.6000	**0.6250**	0.5804	0.5500
XGBoost	0.6000	0.6027	**0.6027**	**0.6000**
LightGBM	0.6000	0.5982	0.5982	0.5982
RandomForest	0.5333	0.5313	0.5313	0.5313
GradientBoosting	0.5000	0.5045	0.5045	0.4994
DecisionTree	0.4667	0.4750	0.4777	0.4570
AdaBoost	0.4333	0.4367	0.4375	0.4327

Árbol de decisión (baseline):
El modelo más simple obtuvo un desempeño limitado con un accuracy de 46.7%. Su bajo recall en la clase positiva (0.31) indica que tiene dificultades para identificar correctamente los días en que el precio de la acción sube. Esto sugiere que, aunque útil como punto de referencia, no es adecuado para tareas de predicción financiera en contextos reales.

Random Forest:
Aprovechando la agregación de múltiples árboles, Random Forest mejoró el desempeño general, alcanzando un accuracy de 53.3% y métricas macro por encima de 0.53. Ofrece predicciones más equilibradas entre clases, aunque sigue lejos de los modelos más sofisticados.

AdaBoost:
Este modelo mostró el peor desempeño entre todos los métodos probados (accuracy: 43.3%). Aunque es robusto para ciertos problemas, su sensibilidad al ruido y la naturaleza secuencial de su entrenamiento lo hacen menos efectivo en este dataset, que incluye cierta volatilidad y ruido inherente al mercado.

Gradient Boosting**:**
Con una precisión macro de 0.5045 y F1 macro cercano a 0.50, este modelo logró un rendimiento aceptable, aunque fue superado por sus versiones modernas. Su mayor ventaja es la flexibilidad y facilidad de implementación, aunque necesita ajustes más cuidadosos para evitar sobreajuste.

LightGBM:
Este modelo sobresale por su equilibrio: accuracy, precisión, recall y f1 macro de 0.60. Es decir, predice correctamente 3 de cada 5 días y lo hace sin sesgo hacia una clase específica. Este balance lo convierte en un candidato sólido para estrategias de trading consistentes, donde se busca minimizar tanto los falsos positivos (entrar al mercado innecesariamente) como los falsos negativos (perder oportunidades reales).

XGBoost:
Con el mejor F1 macro (0.60) y el recall más equilibrado entre clases (0.64 en bajadas, 0.56 en subidas), XGBoost demostró ser el modelo más confiable desde el punto de vista generalista. Su capacidad para detectar tanto subidas como bajadas con alta simetría es ideal para sistemas que operan en ambos sentidos (long y short), y su regularización adicional permite controlar el sobreajuste sin sacrificar precisión.

CatBoost:
Si bien alcanzó la misma accuracy general, se destacó por su alta precisión macro (0.625) y un recall clase 1 de 0.88, lo cual indica que es el modelo más eficaz para detectar subidas en el precio. Esta cualidad es especialmente relevante para diseñar estrategias de compra. No obstante, su capacidad para predecir correctamente bajadas fue limitada (recall clase 0 de 0.29), lo que podría traducirse en más señales falsas de compra.

Final Analysis

Con el fin de evaluar la aplicabilidad práctica de los modelos de clasificación entrenados, se implementó una simulación financiera (*backtesting*) en el conjunto de prueba correspondiente a marzo de 2025. Esta simulación tuvo como objetivo replicar una estrategia de inversión diaria, en la cual se ejecuta una operación de compra únicamente si el modelo predice una subida en el precio de cierre del siguiente día. Se partió de un capital inicial de $1000 USD, y se definió que en cada operación se invertiría el 25% del capital, es decir, $250 USD por trade.

La lógica de la simulación consistió en calcular la rentabilidad potencial acumulada a partir de las decisiones tomadas por cada modelo. Para ello, se utilizó la probabilidad de clase positiva (predict_proba) generada por cada modelo, y se evaluaron distintos umbrales de decisión (threshold) desde 0.40 hasta 0.70. Si la probabilidad de subida superaba el umbral, se generaba una señal de compra. La ganancia o pérdida se calculaba como el rendimiento porcentual entre el precio de cierre del día actual y el del día siguiente, aplicado al monto invertido. Finalmente, se computaron las métricas clave por modelo y umbral: número de operaciones realizadas, operaciones ganadoras y perdedoras, ganancia total y capital final.

Esta metodología permite comparar objetivamente qué modelo no solo predice mejor, sino cuál genera mayor rentabilidad, que es la métrica más relevante en escenarios de inversión; y a partir de ella, se obtuvieron los siguientes resultados:

Modelo	Mejor Threshold	Ganancia Total	Capital Final
LightGBM	0.5	$11.93	$1011.93
XGBoost	0.5	$11.88	$1011.88
LightGBM	0.7	$10.45	$1010.45
LightGBM	0.4	$8.60	$1008.60
GradientBoosting	0.5	$3.58	$1003.58
CatBoost	0.5	-$1.29	$998.71
DecisionTree	0.4	-$31.72	$968.28

Los resultados obtenidos revelan diferencias sustanciales en la capacidad de cada modelo para traducir sus predicciones en decisiones financieras exitosas. De todos los modelos evaluados, LightGBM y XGBoost fueron los únicos que lograron generar una ganancia neta positiva consistente, alcanzando ambos un capital final superior a $1011 USD al utilizar un umbral de 0.5. En particular, LightGBM con threshold 0.5 logró el mejor resultado absoluto, alcanzando $1011.93, producto de 15 operaciones de compra, de las cuales 10 fueron exitosas. Le siguió muy de cerca XGBoost, con una ganancia total de $11.88 y un desempeño muy similar en cuanto a precisión operativa (13 trades, 9 aciertos).

Este resultado es coherente con las métricas de clasificación previamente evaluadas, en donde tanto LightGBM como XGBoost obtuvieron los valores más altos de F1 macro (~0.60), mostrando equilibrio entre precisión y recall. Esto refuerza la idea de que un modelo robusto desde el punto de vista estadístico puede tener impacto directo en la toma de decisiones de trading rentables.

En tercer lugar, Gradient Boosting también generó rentabilidad positiva, aunque de manera más modesta. Con un umbral de 0.5, obtuvo una ganancia de $3.58, demostrando cierta capacidad predictiva, aunque inferior a los métodos más modernos. En contraste, modelos como Decision Tree, AdaBoost y CatBoost mostraron un desempeño deficiente en términos financieros. Decision Tree, por ejemplo, perdió hasta $31.72 con threshold 0.4, resultado consistente con su baja precisión y alta tasa de error en clasificación.

Un caso interesante fue CatBoost, que en el análisis de clasificación mostró una excelente capacidad para detectar subidas (recall = 0.88), pero en la simulación financiera generó múltiples señales de compra no rentables. Esto sugiere que su alto recall vino acompañado de un mayor número de falsos positivos, lo cual, en un entorno de inversión, se traduce en entradas fallidas y pérdidas acumuladas. A pesar de su potencial, este comportamiento hace que CatBoost sea menos recomendable como única base para decisiones de trading, al menos sin un filtro adicional.

Conclusión

En función de los resultados obtenidos, se concluye que LightGBM con threshold 0.5 es el modelo que ofrece el mejor balance entre rendimiento estadístico y retorno financiero, convirtiéndose en el candidato ideal para ser implementado en una estrategia automatizada de inversión en el activo JNJ. Su estabilidad, precisión, y capacidad de adaptación lo hacen adecuado para escenarios de corto plazo como el planteado.

XGBoost también es una excelente alternativa, especialmente si se busca una arquitectura más regularizada y con mayor flexibilidad en el ajuste fino de hiperparámetros. Su rendimiento en la simulación fue prácticamente indistinguible del de LightGBM.

Por el contrario, modelos como Decision Tree, AdaBoost y CatBoost, aunque útiles como referencia o en contextos específicos, no demostraron un desempeño rentable en esta simulación particular y deben utilizarse con precaución en contextos reales de inversión.

Bitcoin

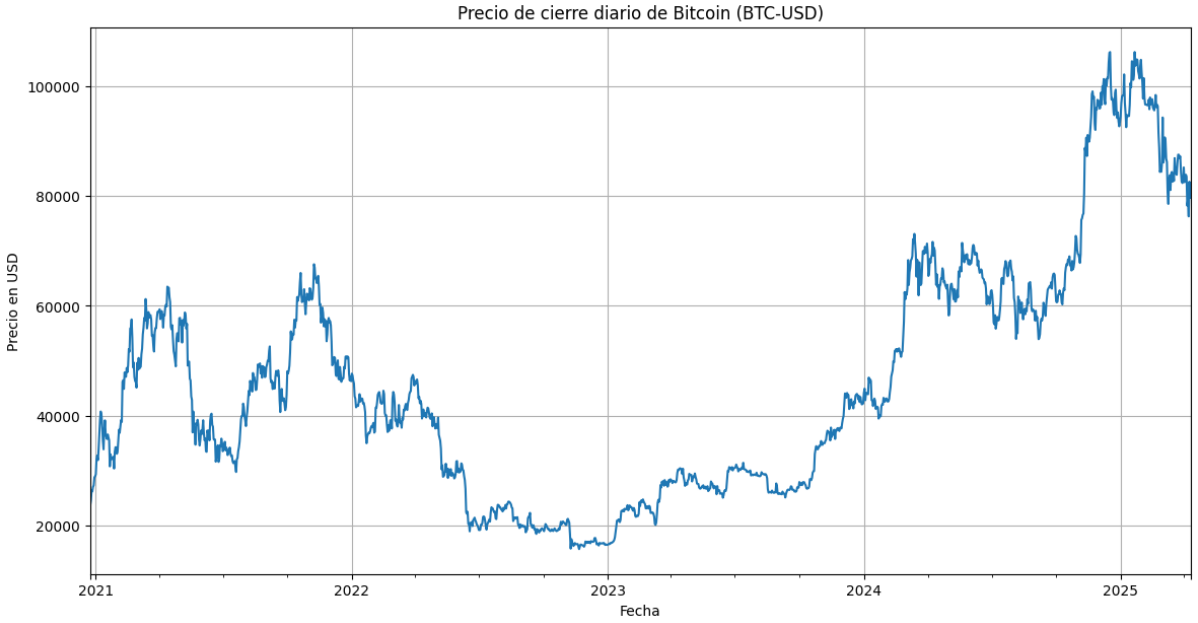
Exploratory Data Analysis (EDA) y Data Wrangling

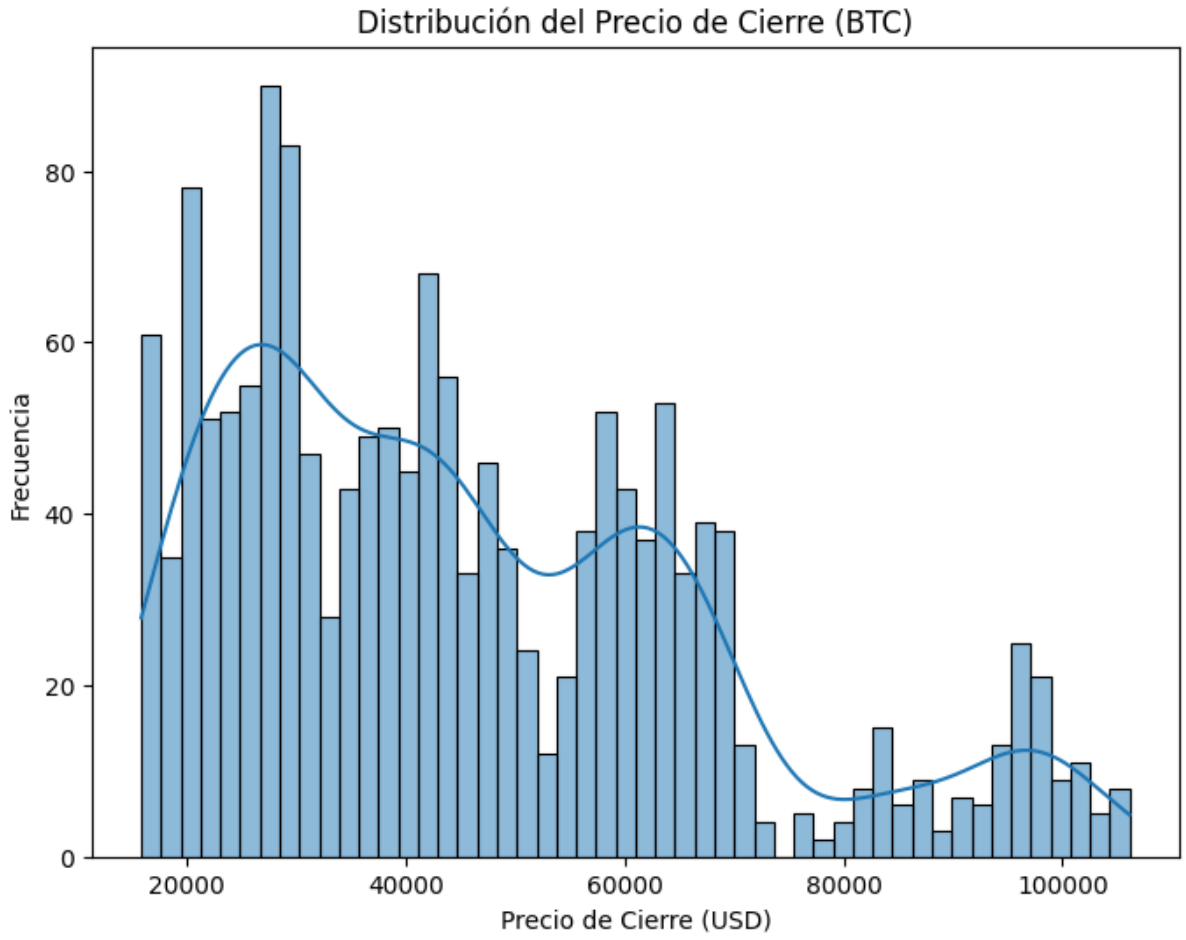
Se utilizaron datos históricos del activo BTC-USD (Bitcoin en dólares estadounidenses) obtenidos desde Yahoo Finance, abarcando el periodo entre el 1 de enero de 2021 y el 1 de abril de 2025. El dataset cuenta con 1570 filas x 5 columnas, sin presencia de valores nulos. Las variables incluidas son: Date, Open, High, Low, Close y Volume.

El análisis estadístico reveló un comportamiento altamente volátil, con precios que oscilaron entre $25,000 y más de $90,000. El volumen también presenta variaciones abruptas, reflejando la naturaleza especulativa del mercado cripto.

El gráfico de precios de cierre diario evidencia una alta volatilidad característica del activo. Se observan ciclos alcistas pronunciados, seguidos de fuertes caídas, especialmente a mediados de 2022 y nuevamente en 2023. A partir de 2024, el precio repuntó fuertemente, superando los $100.000 USD antes de sufrir una corrección reciente.

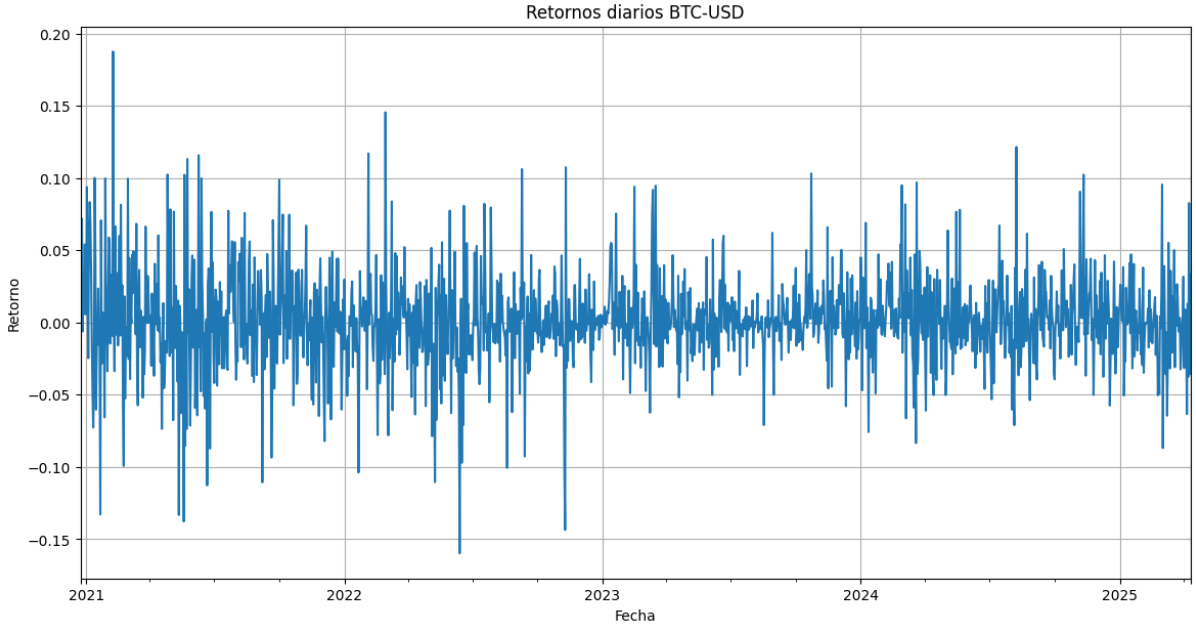
El histograma de precios muestra una distribución multimodal: existen acumulaciones de precios alrededor de los $30.000, $50.000 y $100.000. Esto sugiere periodos prolongados donde el precio permaneció estable, seguidos de cambios abruptos.



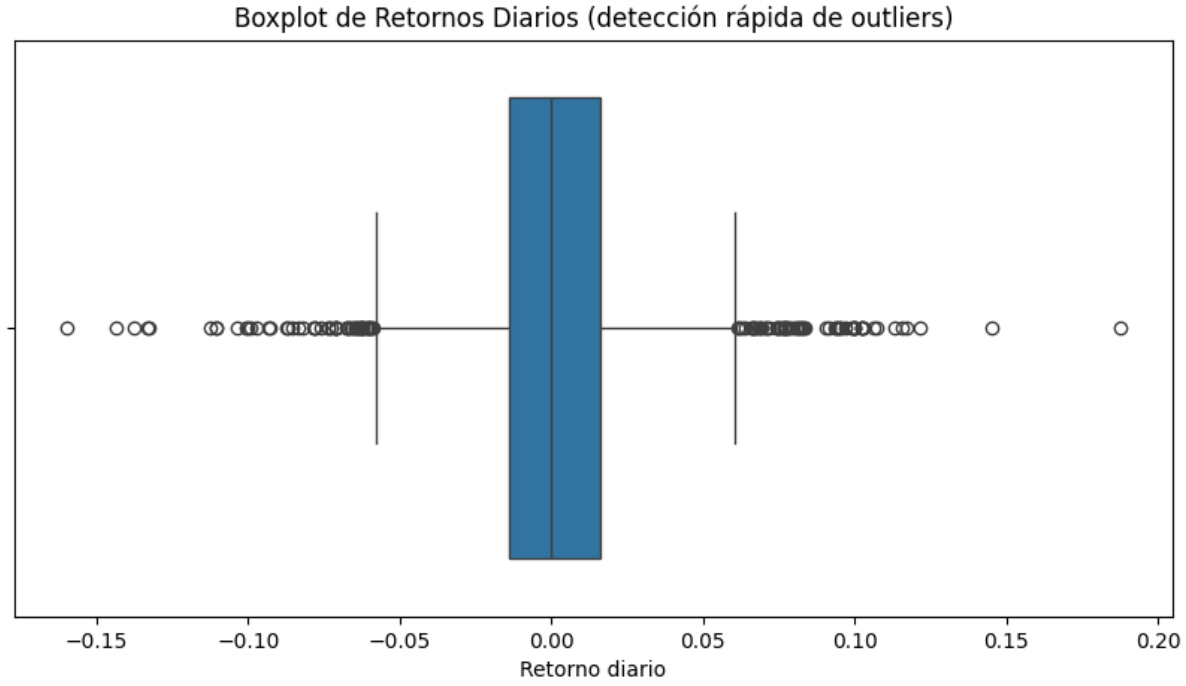


Retornos y comportamiento estadístico

El gráfico de retornos diarios muestra una clara presencia de outliers —tanto positivos como negativos— que reflejan la alta volatilidad del mercado cripto. Aunque la mayoría de cambios están cercanos a cero, no es raro encontrar retornos superiores al 10% o caídas por debajo del -10%.

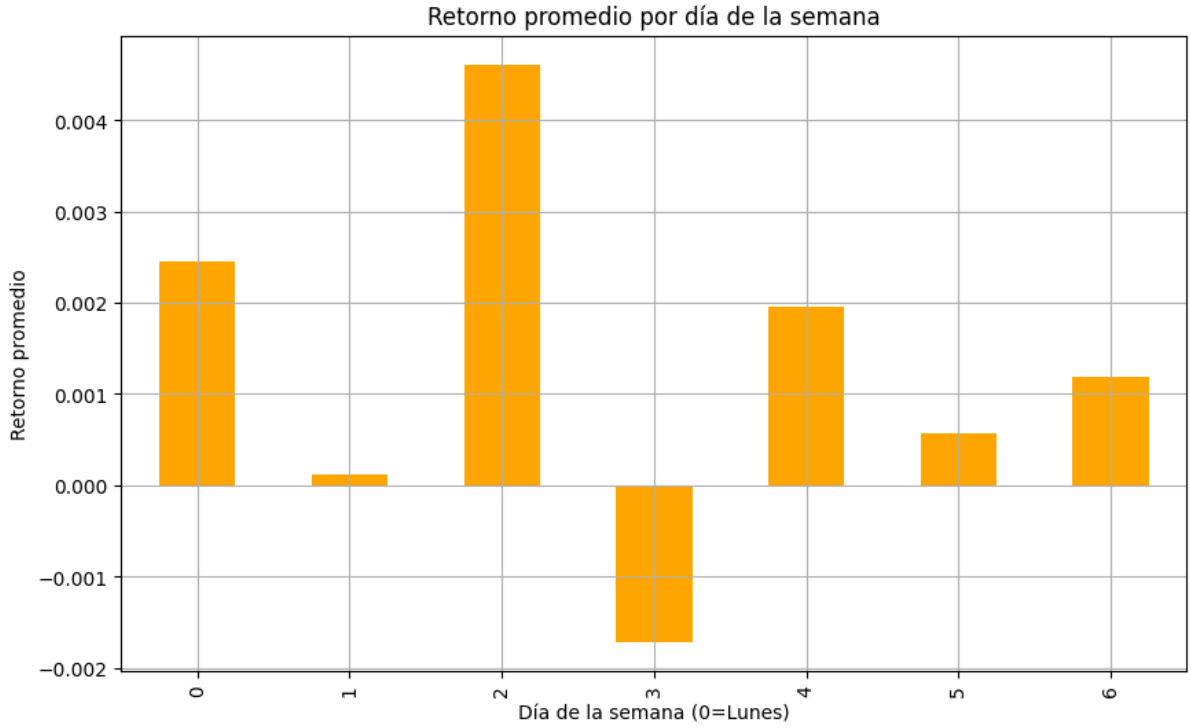


El boxplot confirma este comportamiento: las colas son largas y aparecen muchos puntos atípicos fuera del rango intercuartílico, lo que valida la necesidad de usar modelos robustos para capturar patrones reales sin verse afectados por estos eventos extremos.



Retorno por día de la semana

El análisis de retorno promedio por día de la semana revela que los miércoles tienden a presentar retornos positivos más altos en comparación con otros días. Los lunes y jueves también muestran rendimiento positivo, mientras que los martes y viernes tienen una media más baja o incluso negativa. Esta información temporal puede ser valiosa para la toma de decisiones automatizadas.



Feature Engineering

Para potenciar el desempeño de los modelos de clasificación y permitirles captar patrones relevantes en el precio de Bitcoin, se diseñó un conjunto robusto de variables derivadas que encapsulan tendencias, momentum, volatilidad, volumen y factores temporales. Dado el comportamiento altamente especulativo de BTC, el objetivo fue capturar tanto los ciclos alcistas como bajistas que lo caracterizan.

1. Variables de Precio

- log_return: Retorno logarítmico entre días consecutivos, útil para estabilizar la varianza de los precios y detectar cambios relativos.
- delta_close: Diferencia absoluta del precio de cierre respecto al día anterior, que refleja la magnitud del movimiento diario.

- close_lag1, close_lag2, close_lag3: Precios de cierre desplazados uno, dos y tres días atrás, para proporcionar contexto histórico al modelo.

2. Indicadores de Tendencia

- sma_7, sma_14: Medias móviles simples de 7 y 14 días, ampliamente usadas para detectar tendencias de corto y mediano plazo.
- ema_14: Media móvil exponencial que reacciona más rápidamente a cambios recientes.
- sma_diff: Diferencia entre las medias móviles de corto y mediano plazo (sma_7 - sma_14), útil para detectar cruces de tendencia.
- ema_ratio: Relación entre el precio de cierre y la EMA, útil como medida relativa de momentum.
- close_minus_sma14, close_minus_ema14: Diferencias absolutas entre el precio y las medias móviles, para identificar distancia respecto a la tendencia.

3. Indicadores de Momentum

- macd y macd_signal: MACD (Moving Average Convergence Divergence) y su línea de señal, comúnmente usados para identificar cambios en el momentum.
- rsi_14: Índice de Fuerza Relativa (RSI) de 14 días, que mide condiciones de sobrecompra o sobreventa.
- roc_12: Rate of Change (ROC) de 12 días, indicador de aceleración del precio.

4. Indicadores de Volatilidad

- rolling_std_14: Desviación estándar móvil del precio de cierre (ventana de 14 días), como medida de volatilidad reciente.
- bollinger_width: Ancho de las Bandas de Bollinger, que refleja la amplitud de la variación de precios.

5. Indicadores de Volumen

- volume_log: Transformación logarítmica del volumen para normalizar su distribución.
- volume_sma_5: Media móvil del volumen sobre 5 días.
- obv: On-Balance Volume (OBV), que combina precio y volumen para identificar acumulación o distribución de capital.

6. Variables Temporales

- day_of_week: Día de la semana (0=Lunes, 6=Domingo), para capturar efectos estacionales.

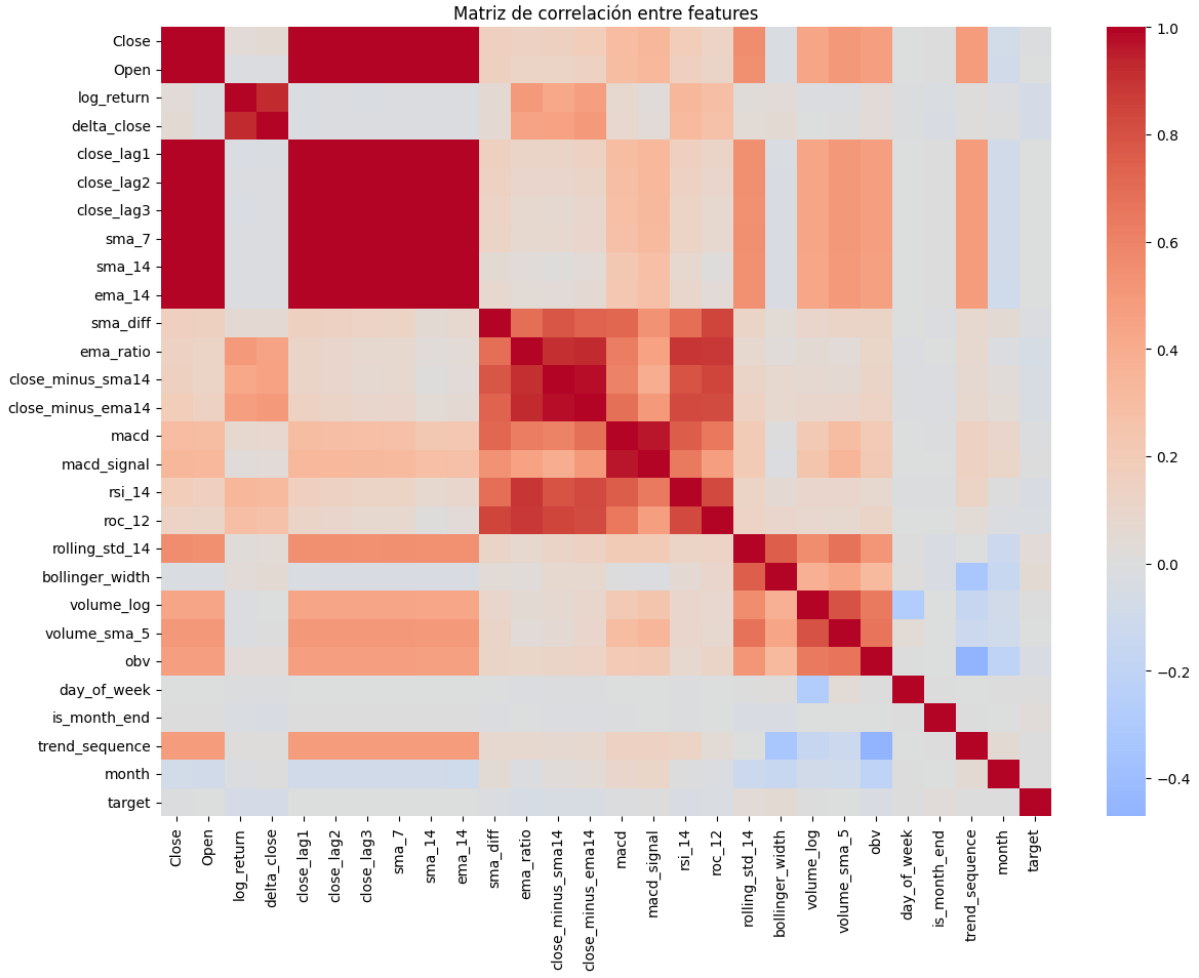
- is_month_end: Indicador binario de fin de mes (1 si es último día hábil del mes, 0 en caso contrario).
- trend_sequence: Contador de días desde el inicio del dataset, útil para modelar comportamientos secuenciales.
- month: Mes del año, para identificar ciclos estacionales anuales.

7. Variable Objetivo

- target: Variable binaria definida como 1 si el precio de cierre del día siguiente es mayor al del día actual, 0 en caso contrario. Representa la señal de compra o no-compra del sistema.

8. Limpieza Final

Todos los cálculos que involucraron ventanas móviles o transformaciones temporales generaron valores NaN en los extremos. Estas filas fueron eliminadas para garantizar un dataset completamente limpio y listo para el modelado.



Selección de Variables

Tras generar una amplia variedad de features técnicas, de volumen, temporales y de tendencia, se aplicó un proceso de **selección de características utilizando SelectKBest**, con el objetivo de reducir la dimensionalidad del modelo y quedarnos únicamente con las variables más informativas para la predicción del target.

Este proceso evalúa la relación estadística entre cada feature y la variable objetivo, seleccionando aquellas con mayor poder predictivo. Como resultado, se obtuvieron las siguientes 20 características seleccionadas:

['Close', 'log_return', 'delta_close', 'sma_14', 'ema_14', 'sma_diff',

'ema_ratio', 'close_minus_sma14', 'close_minus_ema14', 'macd', 'rsi_14',

'roc_12', 'rolling_std_14', 'bollinger_width', 'volume_log', 'volume_sma_5',

'obv', 'day_of_week', 'is_month_end', 'trend_sequence']

Estas variables fueron utilizadas en el entrenamiento de todos los modelos posteriores. Su combinación permite capturar tanto información reciente del mercado como patrones históricos de tendencia, momentum y volatilidad.

Modelado y Evaluación

Tras completar el proceso de ingeniería de características y selección de variables, se entrenaron múltiples modelos de clasificación para predecir la dirección del precio de Bitcoin al siguiente día (target binario: sube o baja). Se utilizaron los siguientes algoritmos:

- Árbol de Decisión
- Random Forest
- AdaBoost
- Gradient Boosting
- LightGBM
- XGBoost

Todos los modelos fueron evaluados utilizando las mismas métricas estándar: **accuracy**, **precision**, **ROC AUC**, **retorno simulado**, **Sharpe ratio**, y **win rate**. Además, se implementó una simulación de trading para estimar la rentabilidad real de las señales generadas por cada modelo, usando un umbral (threshold) conservador de 0.6 para maximizar la confiabilidad de las señales de compra.

Modelos Destacados

Los resultados demostraron que los modelos de boosting modernos fueron notablemente superiores a los métodos tradicionales:

- **LightGBM fue el mejor modelo en términos generales**, con la **mayor precisión (56.25%)**, **mayor retorno simulado ($352.81)** y un **Sharpe ratio sobresaliente de 4.97**, lo que indica una excelente relación entre rentabilidad y riesgo. Su tasa de aciertos también fue la más alta, con un **win rate del 66.67%**.
- **XGBoost ocupó el segundo lugar**, con resultados **muy similares**: precisión de **56.25%**, retorno de **$154.94**, y **Sharpe ratio de 2.43**, manteniendo el mismo nivel de aciertos en las operaciones.

Ambos modelos lograron capturar correctamente señales de trading rentables, con un excelente balance entre precisión y retorno ajustado por riesgo. Su rendimiento confirma la efectividad de los métodos de boosting cuando se aplican a series temporales financieras con alta volatilidad como Bitcoin.