Inteligencia Artificial para la predicción del precio del Bitcoin

Luciano Garrido, Pablo M Angerosa, Diego Altamiranda

Octube 2023

Resumen

This research paper delves into the intricate landscape of Bitcoin trading through a comprehensive exploration of strategies grounded in statistical analysis and machine learning techniques. It presents a detailed analysis of various strategies, including their implementation and evaluation using key metrics such as Accuracy, Cumulative Profit, and PL Ratio (Profit and Loss Ratio). Despite the seemingly random nature of Bitcoin returns, certain strategies demonstrate consistent and profitable patterns, challenging the notion of a completely random market. These promising findings open the door to future research, including the application of more advanced techniques to explore even more complex patterns in the cryptocurrency market.

Keywords: Bitcoin, Trading, Statistical Analysis, Machine Learning, Random Walk, Linear Models, Time Series, Random Forest, Recurrent Neural Network, Cryptocurrencies.

1. Introducción

El mercado de Bitcoin, desde su creación en 2009, ha sido objeto de un interés creciente debido a su extrema volatilidad y su potencial como activo de inversión. Sin embargo, la predicción de los valores del mercado de Bitcoin ha demostrado ser un desafío formidable debido a su comportamiento completamente aleatorio, que se alinea con las premisas del principio de mercado eficiente.[1]

Según este principio, los precios de los activos reflejan toda la información disponible en el mercado, lo que implica que la mejor predicción del precio futuro es el precio actual, a menos que se cuente con información privilegiada.

Este comportamiento de paseo aleatorio ha llevado a la creencia generalizada de que los precios de Bitcoin son esencialmente impredecibles, lo que plantea un desafío significativo para los inversores, traders y analistas. Sin embargo, en este contexto, la tecnología del blockchain nos ofrece una oportunidad para acceder a información que, en otros tipos de activos financieros, no es de fácil acceso.

[1] Hipótesis del Mercado Eficiente: `https://es.wikipedia.org/wiki/Hiptesis_del_mercado_eficiente`.

La cadena de bloques de Bitcoin contiene un registro completo de todas las transacciones realizadas en la red, incluyendo detalles sobre el volumen, la hora y la dirección de origen y destino. Este vasto conjunto de datos ofrece un panorama detallado de la actividad en el mercado de Bitcoin y, potencialmente, patrones ocultos que pueden ayudar a comprender y predecir su comportamiento de precios.

En base lo antedicho, proponemos estimadores que buscan sintetizar la información del blockchain con el fin de superar el comportamiento de paseo aleatorio en la predicción de precios de Bitcoin.

2. Descripción

Las variables utilizadas se extraen de Glassnode[2] , una de las principales herramientas de análisis de blockchain. En una fase inicial, se realizó una selección de 158 variables desde el 2012-01-05 hasta 2023-02-26 en un análisis diario y se agrupan en dos categorías principales. Por un lado, se encuentran las variables inherentes al blockchain, y por otro, aquellas que incorporan información de la cadena de bloques pero están relacionadas con métricas de precios. En el Anexo A del artículo, se detallan todas las variables preseleccionadas. Las tablas del Anexo A, que abarcan desde la Tabla 2 hasta la Tabla 5, incluyen las variables que no contienen ninguna métrica de precios, mientras que la Tabla 6 comprende las variables que están vinculadas al precio.

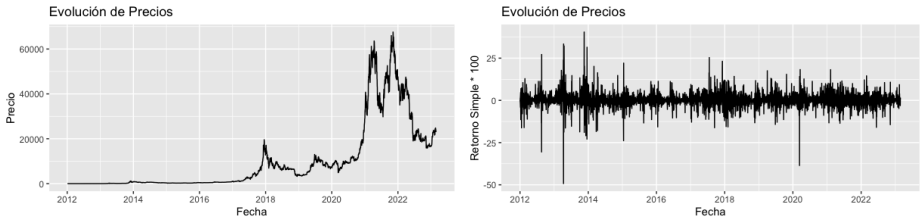
En el análisis financiero, se prefiere estudiar los retornos en lugar de los precios directamente. Esto se debe a que los precios de las acciones a menudo siguen patrones exponenciales difíciles de manejar matemáticamente. Los rendimientos, por otro lado, son una medida más efectiva, ya que tienen propiedades estadísticas más manejables, como la estacionariedad y la ergodicidad. Si bien hay muchas variantes de la definición de retorno, dependiendo de otras variables en su cálculo, como los dividendos o los costos de transacciones, en este estudio se decide utilizar el enfoque mas simple de retorno que se muestra a contiunación en la Definición 2

Definición 1. ***(Retorno Simple[1])*** *Sea P_t el precio de un activo en el tiempo t. Dada una escala de tiempo τ, el retorno simple de período τ en el tiempo t, $R_t(\tau)$, es la tasa de cambio en el precio obtenida al mantener el activo desde el tiempo $t-\tau$ hasta el tiempo t:*

$$R_t(\tau) = \frac{P_t - P_{t-\tau}}{P_{t-\tau}} = \frac{P_t}{P_{t-\tau}} - 1$$

En el Gráfico 1 se puede observar el impacto de aplicar la transformación del retorno simple a la evolución del precio, de este modo se deshace la característica

[2]Glassnode `https://glassnode.com/`.



(a) Precio (b) Retorno Simple

Gráfico 1: Comparación entre la evolución de precios y la evolución del retorno simple desde 2012 hasta 2023

exponencial presente en la evolución del precio. Siguiendo a esto, en nuestro estudio denominamos esta transformación como *stock_return*.

2.1. BTC Como Paseo Aleatorio

A consecuencia de la hipótesis de mercado eficiente[1], podemos concluir que el BTC se comporta como un paseo aleatorio. Para un paseo aleatorio, la mejor predicción a un paso es simplemente el valor actual del proceso. En otras palabras, en un paseo aleatorio, no se puede hacer una predicción mejor que decir que el siguiente valor será igual al valor actual, ya que se supone que los movimientos son aleatorios e impredecibles.

Un paseo aleatorio es un proceso estocástico en el que cada nuevo valor es el resultado de una variable aleatoria, y estos valores no están correlacionados entre sí ni siguen un patrón discernible. Esto significa que cualquier intento de predecir el próximo valor no tiene una base sólida y no se puede hacer mejor que simplemente asumir que será igual al valor actual.

En el contexto financiero, la teoría del paseo aleatorio sugiere que los precios de los activos financieros, como las acciones, son impredecibles a corto plazo y que no se pueden predecir de manera confiable utilizando análisis técnico o tendencias históricas a corto plazo. Esta es una idea fundamental en la teoría de los mercados eficientes, que sostiene que los precios de los activos reflejan toda la información disponible y que los movimientos de precios futuros son esencialmente impredecibles.

Como una muestra de esto al realizar la serie de tiempo de *stock_returns*, el algoritmo de selección automática AUTO.ARIMA [3] determina que el mejor modelo de ajuste es un ARIMA(0,0,0) En otras palabras, la serie de tiempo no muestra patrones de tendencia ni de estacionalidad significativos y que los valores futuros no están relacionados de manera significativa con los valores pasados.

Parte del objetivo de este estudio es encontrar la metodología para tratar de romper el paseo aleatorio ya sea con información privilegiada de la cadena de

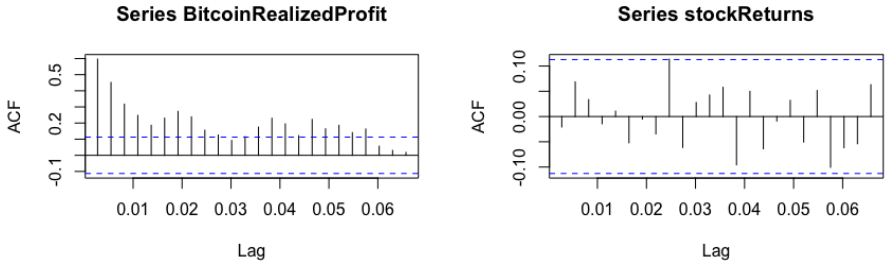
[3] Auto.Arima `https://pkg.robjhyndman.com/forecast/reference/auto.arima.html`.

bloques o transformaciones del precio que incorporan esta información.

Por ejemplo la variable *Bitcoin Realized Profit USD* que se refiere a una métrica que se utiliza en el análisis de Bitcoin para estimar las ganancias o pérdidas realizadas por sus inversores en función del precio al que compraron sus monedas y el precio actual de mercado. Esta métrica proporciona una perspectiva sobre cuánto dinero ganaron o perdieron los inversores que vendieron sus Bitcoin en un momento específico.

La fórmula básica para calcular el *Bitcoin Realized Profit USD* es la siguiente: *Bitcoin Realized Profit USD = Cantidad de Bitcoin vendidos × (Precio de venta - Precio de compra)*

Al aplicar la función de autocorrelación y examinar su gráfico en una ventana de tiempo dada, se observa claramente que existe correlación entre los rezagos, en contraste con lo que ocurre en el caso de *stock_returns*, como se detalla en la Gráfica 2



(a) BTC Realized Profit ACF

(b) Stock Returns ACF

Gráfico 2: Comparación entre la función de autocorrelación de BTC Realized Profit y la función de autocorrelación de Stock Returns.

Debido a esto, al aplicar la función 'auto.arima' a cada una de las series, se observa que la serie *stock_returns* se ajusta mejor a un modelo ARIMA(0,0,0) en esta ventana de tiempo. En contraste, la serie *Bitcoin Realized Profit USD* sugiere un modelo de ajuste de ARIMA(2,1,2), a pesar de que esta variable incluye implícitamente el precio del BTC.

Esto sugiere que al incorporar información privilegiada al modelo, podría existir la posibilidad de romper el comportamiento de paseo aleatorio, lo cual es la motivación y enfoque de este estudio.

3. Metodología

3.1. El modelo

La premisa central de este modelo es lograr una explicación del precio del Bitcoin a través de una relación lineal entre variables, que, en su propia serie de

tiempo, no exhiba un comportamiento de paseo aleatorio. Una vez que hayamos identificado el modelo que mejor se ajusta a los datos dentro de una ventana de tiempo específica, procederemos a predecir el próximo valor de cada una de las variables, considerándolas como series de tiempo, utilizando diversas técnicas de aprendizaje automático. Entre estas técnicas se incluyen el Análisis de Series Cronológicas, el Análisis de Series Cronológicas Multivariadas, Bosques Aleatorios y Redes Neuronales Recurrentes.

Después de obtener las predicciones para cada una de estas variables tratadas como series de tiempo, las incorporaremos en el modelo lineal seleccionado para obtener la predicción del precio para el próximo valor del Bitcoin. En el Gráfico 3 se detalla el funcionamiento del modelo. Primariamente se selecciona una venta de entrenamiento con todo el conjunto de variables. De esta ventana mediante el algoritmo de selección de variables StepWise, se seleccionan las variables que mejor explican linealmente al retorno simple, en esta ventana de tiempo. Cada una de estas variables seleccionadas, en el ejemplo del diagrama x_1, x_2, x_4, x_7 se tratan como una serie de tiempo en una nueva ventana de entrenamiento para cada una de ella. Antes de realizar la predicción de estas series, realizamos iterativamente los tests necesarios para verificar que la serie es estacionaria (media constante, homocedasticidad y covarianzas entre rezagos que solo dependen del rezago y no del tiempo). Una vez verificados estos supuestos, se realiza la predicción para el siguiente intervalo de tiempo $t + 1$, con técnicas que algunas consideran el problema desde una perspectiva multivarada y otras como series independientes. Se evalúa con estas predicción el modelo lineal seleccionado por el StepWise, obteniendo asi la prediccion del retorno simple para el momento $t + 1$

3.2. Benchmark: El Estimador Naive

Un benchmark es un punto de referencia esencial utilizado para evaluar y comparar la eficacia de varios estimadores propuestos. En el contexto de condiciones de paseo aleatorio, como hemos mencionado previamente, la estimación base se basa en el último valor observado. Este último valor actúa como el estimador principal para la evaluación de otros métodos propuestos, lo que permite determinar si estos métodos son más efectivos en la estimación de la variable de interés o si muestran un desempeño comparable al enfoque de referencia.

La función principal de un benchmark en este contexto es servir como un estándar de comparación que facilita la identificación de estimadores alternativos que puedan ofrecer mejoras significativas en la precisión o la eficiencia de la estimación en comparación con el último valor observado. La elección del último valor como estimador base se basa en la suposición de que en condiciones de paseo aleatorio, el valor más reciente tiende a ser un buen predictor del valor futuro.

$$\hat{Y}_{t+1} = Y_t$$

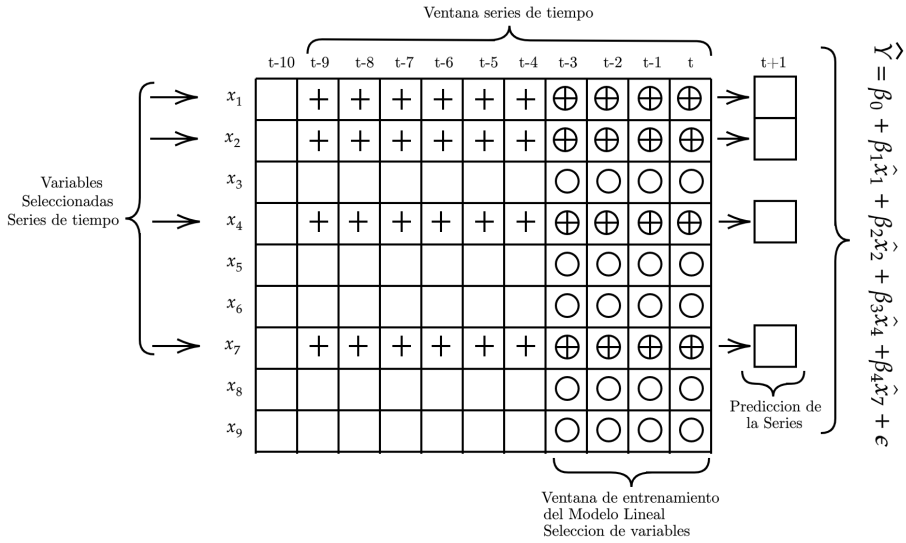


Gráfico 3: Descripción gráfica del modelo. Comportamiento de las ventanas de entrenamiento

3.2.1. Performance del estimador Naive en las variables X como series de tiempo univaradas

Con el propósito de evaluar el desempeño del estimador naive, llevamos a cabo el siguiente proceso. Para cada una de las variables X, consideradas como series temporales, desarrollamos dos enfoques de estimación dentro de una ventana de tiempo definida. El primero de estos enfoques empleó el estimador naive, mientras que el segundo se basó en la aplicación de AUTO.ARIMA a la serie univariable correspondiente.

Este proceso de evaluación se extendió durante un año completo. Los resultados obtenidos indican que el estimador naive no siempre supera las predicciones generadas por el modelo AUTO.ARIMA seleccionado, lo cual podiamos intuir cuando mencionamos previamente, en las secciones anteriores que el auto.arima no simpre elige como mejor ajuste el Modelo ARIMA(0,0,0) para las variables X

3.3. Modelo Lineales

Los modelos lineales son un conjunto de técnicas estadísticas utilizadas en el análisis de datos que buscan establecer una relación lineal entre una variable dependiente y una o más variables independientes. Estos modelos se emplean para comprender y predecir la variación en la variable dependiente en función de las variaciones en las variables independientes.

En el contexto de este análisis, los modelos lineales se utilizan primariamen-

te para analizar y describir el comportamiento del retorno simple, en relación con otras variables. Este modelos se aplica dentro de una ventana de tiempo específica, lo que significa que se analiza un período determinado de datos.

El proceso de construir y ajustar un modelo lineal implica encontrar la mejor relación lineal posible entre las variables independientes y la variable de interés, de modo que pueda describirse de manera precisa y cuantitativa cómo los cambios en las variables independientes afectan los cambios en la variable dependiente (en este caso, el retorno simple). Los modelos lineales también pueden proporcionar coeficientes que indican la magnitud y la dirección de la influencia de cada variable independiente en el retorno simple.

La esencia de utilizar modelos lineales en este estudio radica en la necesidad de desglosar y comprender el comportamiento del retorno simple en el BTC.

$$Y_t = \vec{\beta}\mathbf{X}_t + \epsilon \tag{1}$$

Notar que en la Ecuación 1 la variable explicada no es una predicción sino que es la relación lineal de variables que mejor explica al retorno simple en una ventana de tiempo dada.

3.3.1. Stepwise

El algoritmo de selección de variables Stepwise es una técnica utilizada en estadísticas y análisis de datos para construir modelos de regresión lineal o logística seleccionando de manera automática las variables independientes más relevantes. El objetivo es simplificar el modelo al eliminar variables que no aportan significativamente a la explicación de la variable dependiente, lo que puede mejorar la interpretación y la eficiencia del modelo.

La definición del algoritmo de selección de variables Stepwise puede ser la siguiente:

El algoritmo Stepwise es un procedimiento iterativo que comienza con un conjunto inicial de variables independientes y luego realiza una serie de pasos para agregar o eliminar variables en función de su contribución estadística al modelo. Hay dos enfoques principales en el algoritmo Stepwise: hacia adelante (forward) y hacia atrás (backward). En el enfoque hacia adelante, el algoritmo comienza con un modelo vacío y agrega una variable a la vez, seleccionando la que mejora el modelo de manera más significativa. En el enfoque hacia atrás, comienza con un modelo que incluye todas las variables y elimina una a la vez, excluyendo la que tiene la menor relevancia estadística. El proceso continúa hasta que se cumple un criterio de parada predefinido, como un valor de p-value o una métrica de rendimiento específica. El resultado final es un modelo simplificado que incluye solo las variables que se consideran más relevantes para la predicción de la variable dependiente.

Los parámetros de selección de variables *alpha_remove* y *alpha_enter* en el algoritmo Stepwise son valores umbral que determinan cuándo una variable debe ser eliminada (remove) o agregada (enter) al modelo. *alpha_remove* establece el nivel de significación necesario para que una variable existente sea eliminada del

modelo, mientras que *alpha_enter* establece el nivel de significación requerido para que una nueva variable sea incluida en el modelo. Un valor más bajo para estos alphas hace que el algoritmo sea más estricto en la selección de variables, lo que resulta en modelos más simples con menos variables. En contraste, valores más altos permiten una mayor flexibilidad en la inclusión o eliminación de variables, lo que puede conducir a modelos más complejos.

3.4. Series Cronologicas

Una vez que las variables han sido seleccionadas mediante el algoritmo Stepwise, se avanza en el proceso de análisis al tratar estas variables como series de tiempo en una nueva ventana de entrenamiento. Las series de tiempo son conjuntos de observaciones registradas en intervalos regulares a lo largo del tiempo. En este contexto, se utilizan para analizar y modelar la evolución de las variables a lo largo de una secuencia temporal, identificando patrones, tendencias y estacionalidades.

3.4.1. Estacionariedad

En el análisis de series temporales, es importante que los datos sean estacionarios, lo que significa que sus propiedades estadísticas no cambian con el tiempo. La estacionariedad débil es una propiedad importante en el análisis de series temporales y se refiere a que las propiedades estadísticas de una serie temporal son aproximadamente constantes a lo largo del tiempo.

Las principales propiedades estadísticas de la estacionariedad débil son las siguientes:

Media constante: Significa que la media de la serie temporal es constante a lo largo del tiempo. Esto implica que la serie no tiene una tendencia sistemática a aumentar o disminuir con el tiempo.

Varianza constante: La varianza de la serie temporal también es constante a lo largo del tiempo. Esto significa que la dispersión de los datos alrededor de la media no cambia con el tiempo.

Autocorrelación constante: La autocorrelación de la serie, que mide la relación entre los valores de la serie en diferentes momentos en el tiempo, es constante para todos los desplazamientos temporales. En otras palabras, la dependencia entre los valores de la serie en diferentes momentos en el tiempo no cambia con el tiempo.

Si una serie no es estacionaria, se puede aplicar diferenciación para convertirla en estacionaria antes de aplicar modelos ARIMA.

3.4.2. Test Dicky-Fuller y Diferenciación de la Serie

El test de Dickey-Fuller es una prueba estadística que se utiliza para evaluar la presencia de raíces unitarias en una serie temporal, lo que está relacionado con la estacionariedad de la serie.

La raíz unitaria es una característica de una serie temporal en la que una serie muestra una fuerte dependencia de su propio pasado, lo que puede llevar a la no estacionariedad de la serie. En otras palabras, una serie con una raíz unitaria no cumple con las propiedades estadísticas de la estacionariedad débil que mencionamos anteriormente.

El test de Dickey-Fuller se utiliza para probar la hipótesis nula (H0) de que una serie temporal tiene una raíz unitaria, lo que significa que la serie es no estacionaria. La hipótesis alternativa (H1) es que la serie no tiene una raíz unitaria y, por lo tanto, es estacionaria.

Se aconseja diferenciar una serie temporal si no pasa el test de Dickey-Fuller porque la diferenciación es una técnica comúnmente utilizada para convertir una serie no estacionaria en una serie estacionaria.

La idea detrás de la diferenciación es eliminar la tendencia o la dependencia de los valores anteriores para que la serie resultante sea más fácil de analizar y modelar.

Después de aplicar la diferenciación, es posible que la nueva serie resultante pase el test de Dickey-Fuller, lo que indica que es estacionaria.

En nuestro caso, para cada ventana de tiempo de las X y antes de estimar el modelo lineal, realizamos de forma iterativa el test de Dicky Fuller. En caso de que los resultados del test indiquen la necesidad de realizar ajustes para lograr la estacionariedad de la serie, procedemos a aplicar la técnica de diferenciación de manera iterativa en cada ciclo hasta que la serie cumpla con los requisitos del test.

3.4.3. Predicciones

El objetivo principal del análisis de series temporales es hacer predicciones precisas sobre valores futuros. Para ello utilizamos enfoques estadisticos y de machine learning. El objetivo es lograr una prediccion a un paso $\hat{\mathbf{X}}_{t+1}$ de cada una de las variables seleccionadas, vistas como series.

3.4.3.1. Series Cronologicas Univariadas Para comprender y predecir el comportamiento de una serie temporal univariada, se utilizan técnicas de análisis y modelado, como el modelado ARIMA . Los modelos ARIMA (Auto-Regressive Integrated Moving Average) son herramientas estadísticas ampliamente utilizadas en el análisis de series temporales. Estos modelos combinan tres componentes principales: autorregresión (AR), integración (I) y promedio móvil (MA). El componente AR refleja la dependencia de un valor actual en valores previos de la serie temporal, el componente I implica diferenciación para lograr estacionariedad y el componente MA utiliza errores pasados para hacer predicciones.

ARIMA(p, d, q) :

$$\left(1 - \phi_1 L - \phi_2 L^2 - \ldots - \phi_p L^p\right)(1 - L)^d X_t = \left(1 + \theta_1 L + \theta_2 L^2 + \ldots + \theta_q L^q\right)\varepsilon_t$$

3.4.3.2. Series Cronologicas Multivariadas Los modelos VAR (Vector AutoRegressive) son una clase de modelos econométricos utilizados para analizar y pronosticar múltiples series temporales que están interrelacionadas entre sí. En lugar de enfocarse en una sola variable, como en los modelos univariados, los modelos VAR consideran conjuntamente varias variables y modelan sus relaciones a lo largo del tiempo. Estos modelos capturan las dependencias mutuas entre las series, lo que permite analizar cómo los cambios en una variable afectan a las otras y cómo evolucionan conjuntamente en el tiempo. La fórmula matemática de un modelo VAR (Vector AutoRegressive) para dos series temporales, por ejemplo, y_1 y y_2, se expresa de la siguiente manera:

$$y_{1,t} = c_1 + \phi_{11,1} y_{1,t-1} + \phi_{12,1} y_{2,t-1} + \varepsilon_{1,t}$$
$$y_{2,t} = c_2 + \phi_{21,2} y_{1,t-1} + \phi_{22,2} y_{2,t-1} + \varepsilon_{2,t}$$

3.4.3.3. Bosques Aleatorios Los Bosques Aleatorios para Series Temporales (Random Forests for Time Series) son una extensión de los Bosques Aleatorios tradicionales diseñada para abordar problemas de pronóstico en series temporales. A diferencia de los Bosques Aleatorios convencionales, donde las observaciones son independientes entre sí, en series temporales, las observaciones están correlacionadas en función del tiempo. Los Bosques Aleatorios para Series Temporales aplican estrategias específicas para lidiar con esta correlación temporal. Estos modelos pueden ser útiles para predecir valores futuros en series temporales, y su capacidad para manejar dependencias temporales los hace valiosos en aplicaciones como la predicción de ventas, el pronóstico meteorológico y otros escenarios en los que se necesita tener en cuenta la estructura temporal de los datos.

3.4.3.4. Redes Neuronales Recurrentes Las Redes Neuronales Recurrentes (RNN, por sus siglas en inglés) son un tipo de arquitectura de redes neuronales artificiales diseñadas específicamente para el procesamiento de datos secuenciales, como series temporales, texto o señales de audio. A diferencia de las redes neuronales feedforward tradicionales, las RNN tienen conexiones recursivas que les permiten mantener una memoria interna y procesar secuencias de longitud variable. Cada neurona en una RNN toma una entrada y genera una salida, y además mantiene un estado oculto que captura la información de las entradas anteriores. Esto les permite modelar dependencias temporales y capturar patrones en datos secuenciales.

$$h_t = f\left(W_{hh} h_{t-1} + W_{xh} x_t + b_h\right)$$

- h_t es el estado oculto o la activación de la neurona en el momento t.
- h_{t-1} es el estado oculto en el momento anterior $t-1$.
- x_t es la entrada en el momento t.
- W_{hh} es la matriz de pesos que representa las conexiones recurrentes entre las neuronas en el mismo instante de tiempo t.

- W_{xh} es la matriz de pesos que representa las conexiones entre las entradas x_t y el estado oculto h_t.

- b_h es el sesgo o bias de la neurona.

- f es la función de activación, que puede variar según el tipo de RNN (por ejemplo, sigmoide, tangente hiperbólica o unidad lineal rectificada).

3.5. Estrategia y Medidas de Rendimiento

3.5.1. Estrategia

La estrategia implementada se basa en un enfoque sorprendentemente simple y efectivo. Primero, se realiza una predicción puntual del retorno simple, es decir, se pronostica cuánto variará el precio desde el momento t hasta el momento $t+1$. De esta predicción, lo único que se retiene es el signo, lo que indica en qué dirección se moverá el mercado durante ese intervalo.

$$\widehat{tendencia} = sign(\widehat{stock\_returns})$$

En esta estrategia de trading, el enfoque es claro y directo. Cuando la estimación del mercado es positiva, el bot entra en una posición long (compra) y, por otro lado, cuando la estimación es negativa, se abre una posición short (venta) y cierra las operaciones cuando termina la vela. Este proceso se repite para cada vela diaria, lo que implica que el bot opera ininterrumpidamente los 365 días del año.

Es importante destacar que en esta estrategia no se utilizan stop loss ni take profit, lo que significa que las operaciones se mantienen abiertas sin un límite predeterminado de pérdidas o ganancias. Esta elección esta basada en que ningún elemento externo al estimador afecte su rendimiento.

La simplicidad de esta estrategia radica en su objetivo principal: obtener una medida estándar para evaluar el rendimiento de los estimadores y poder compararlos de manera efectiva.

3.5.2. Accuracy

El *accuracy* (exactitud) es una métrica comúnmente utilizada para evaluar el rendimiento de un modelo. Se trata de una medida de qué tan bien un modelo clasifica correctamente las instancias de un conjunto de datos.

En el contexto de los estimadores estadísticos se refiere a la proporción de predicciones correctas realizadas por el modelo en comparación con el número total de predicciones. Se calcula mediante la fórmula:

$$Accuracy = \frac{Predicciones\_Correctas}{Predicciones\_Totales}$$

3.5.3. Profit Acumulado

El concepto de "profit acumuladorepresenta el porcentaje ganado al invertir 1 BTC en cada operación realizada. También puede entenderse como la variación

acertada en el retorno de las acciones, predicha por el modelo. Este indicador es crucial para evaluar el rendimiento a largo plazo de una estrategia de trading. Al invertir consistentemente 1 BTC en cada operación y calcular el beneficio acumulado, se obtiene una medida directa de la efectividad del modelo en la predicción de movimientos de precios.

Este enfoque proporciona una visión clara del rendimiento de la estrategia, ya que se basa en inversiones consistentes y muestra cuánto se ha ganado en comparación con la inversión inicial. Además, al comparar el profit acumulado con el rendimiento del mercado o con otras estrategias, se puede determinar la efectividad relativa del modelo en la toma de decisiones de trading.

Al evaluar esta métrica junto con otras, los traders pueden tomar decisiones más informadas sobre la efectividad de sus estrategias y ajustar sus enfoques en consecuencia para optimizar sus ganancias.

3.5.4. Profit and Loss Ratio

El $PL - Ratio$ (Profit and Loss Ratio), también conocido como $Reward - to - RiskRatio$ (Ratio de Recompensa a Riesgo) es una métrica utilizada para evaluar el equilibrio entre el potencial de ganancias y el riesgo asumido en una operación.

Se calcula dividiendo la ganancia potencial (o recompensa) de una operación entre la pérdida potencial (o riesgo) de la misma operación. Esta relación ayuda a evaluar si la estrategia tiene el potencial de generar suficientes ganancias para justificar el riesgo asumido.

$$PL\_Ratio = \frac{Ganancia\_Potencial}{Perdida\_Potencial}$$

Un PL Ratio mayor a 1 indica que la recompensa potencial es mayor que el riesgo, lo que generalmente se considera una buena oportunidad de trading. Un PL Ratio inferior a 1 indica que el riesgo es mayor que la recompensa, lo que puede indicar una operación menos favorable desde el punto de vista del riesgo-recompensa.

4. Implementación y Resultados

4.1. Estimador 1

Inicialmente, realiza una selección de variables utilizando el procedimiento stepwise, el cual se emplea para construir un modelo lineal destinado a describir las relaciones con el precio. Una vez completada la etapa de selección de variables y estimación de coeficientes del modelo, se procede a tratar cada variable seleccionada como una serie de tiempo univariada. Para cada una de estas series temporales, se efectúa una predicción a un paso, es decir, de t a $t + 1$, utilizando modelos SARIMA. Las predicciones resultantes para cada variable son, posteriormente, integradas en la evaluación del modelo lineal. A partir de esta evaluación, se obtiene una predicción del precio en el momento $t + 1$

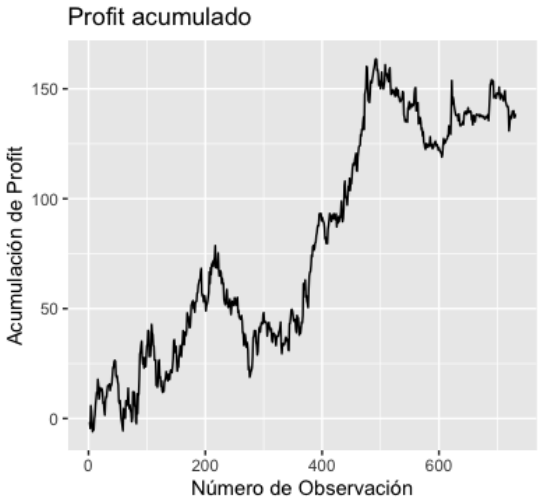


Gráfico 4: Rendimientos del estimador 1 en el periodo de 2 años

En el contexto de este estudio, se realiza el análisis de selección de variables mediante modelos lineales en una ventana móvil de 20 días, lo que implica que el modelo se entrena y selecciona variables dentro de esta ventana utilizando datos desde $t-19$ hasta t. Además, las series de tiempo de las variables seleccionadas se entrenan en una ventana de 300 días hasta el momento t.

Se aplicó el estimador durante un período de 2 años, y los resultados obtenidos sugieren la posibilidad de desafiar la hipótesis del mercado eficiente. Como se puede observar en el Gráfico 4 , el rendimiento del estimador muestra una tendencia claramente alcista, alcanzando resultados altamente positivos en condiciones de operación especiales, especialmente cuando las tarifas proporcionadas por los intercambios son bajas, con un rendimiento final del 138.64 %.

En la Tabla 1 , se observa que el estimador logró un índice de precisión del 54 %, superando la desafiante barrera del 50 %.

Estimador	Accuracy	Profit Acumulado	PL Ratio
Estimador 1	0.5431	153.3	1.16
Estimador 2 (y)	0.4952	2.67	1.003
Estimador 2 (negative x)	0.5212	233.96	1.295
Estimador 3	0.45	-91.43	0.87
Estimador 4	0.50	-50.7	0.94

Cuadro 1: Resultados de los 4 estimadores

4.2. Estimador 2

4.2.1. Y multivariada

En el período de 2 años, se implementó el Estimador 2 utilizando la técnica de series multivariadas VAR. La lógica detrás de este estimador es parcialmente diferente. En este caso, se realiza una selección de variables a través del modelo de stepWise y se analiza el *stock_returns* como una serie multivariada donde la idea que subyace al análisis multivarado es que cada una de estas variables se influyen mutuamente y siguen un patrón conjunto.

La predicción del rendimiento simple se obtiene directamente del análisis de estas series multivariadas, sin la necesidad de aplicar el modelo lineal generado por el stepWise.

Los resultados obtenidos bajo este enfoque se detallan en la Tabla 1. El estimador logra un Accuracy del 0.4952, un profit acumulado de 2.67 y un PL-ratio de 1.003.

Es importante observar el Gráfico 5, donde se muestra la tendencia del profit acumulado. Se puede notar que esta tendencia es errática y no sigue un patrón definido. A lo largo de algunos meses, el rendimiento se duplica, pero luego entra en un ciclo de pérdidas negativas hasta estabilizarse con un profit positivo al final del período analizado.

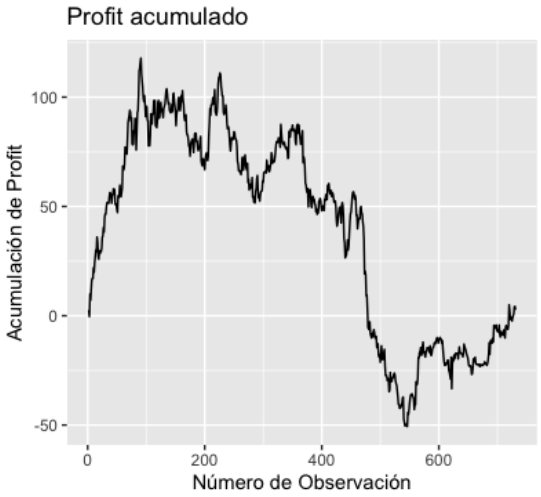


Gráfico 5: Rendimientos del estimador 2 (y) en el periodo de 2 años

4.2.2. Negative X multivariada

En este estimador, se regresa a la estrategia principal que implica el uso de un modelo lineal. Primero, identificamos el modelo lineal mediante StepWise y

luego aplicamos un estudio de series multivariadas para las variables seleccionadas en dicho modelo.

Una vez que hemos realizado la predicción para las variables independientes (X) del modelo lineal, incorporamos estas predicciones de las X nuevamente en el modelo lineal. Posteriormente la estimación obtenida del modelo se somete a una transformación lineal negativa, que ajusta y modifica las predicciones. Esta transformación es un paso crucial para mejorar la precisión del modelo y alinear las predicciones con los patrones observados en los datos históricos.

Los resultados obtenidos bajo este enfoque se detallan en la Tabla 1. El estimador logra un Accuracy del 0.5212, un profit acumulado de 233.96 y un PL-ratio de 1.295.

En el Gráfico 5, se muestra que la tendencia del profit acumulado indica un crecimiento sostenido.

Gráfico 6: Rendimientos del estimador 2 (negative x) en el periodo de 2 años

Es relevante enfocarse en las observaciones de este estimador y sus medidas de rendimiento. A pesar de tener una precisión del 52 %, lo que indica que acierta aproximadamente la mitad del tiempo, el análisis del profit acumulado revela un crecimiento constante. Esto sugiere que el modelo muestra un acierto notorio en las velas con movimientos significativos del mercado, mientras que puede equivocarse en las situaciones donde el mercado tiene fluctuaciones más pequeñas.

Este patrón indica que el modelo está capturando eficientemente las tendencias en las velas con cambios bruscos, posiblemente debido a su capacidad para identificar patrones en movimientos de precios más grandes. Por otro lado, en contextos de volatilidad baja, donde las variaciones son menores, el modelo puede no ser tan preciso, lo que se refleja en el accuracy del 52

4.3. Estimador 3

Para el Estimador 3 inicialmente, se emplea una ventana de tiempo móvil y se aplica modelos lineales junto con el método StepWise para seleccionar las variables. Una vez que se obtienen las variables seleccionadas, se sigue un procedimiento ligeramente diferente para entrenar el modelo usando la técnica de Random Forest.

En este caso, se entrena el modelo con las variables X en el tiempo t y la variable Y (retorno simple) en el tiempo t. Esto permite predecir directamente el retorno simple sin realizar las predicciones de las variables X. A diferencia de los otros estimadores, tanto los modelos lineales como los Bosques Aleatorios se entrenan utilizando ventanas de 4 días.

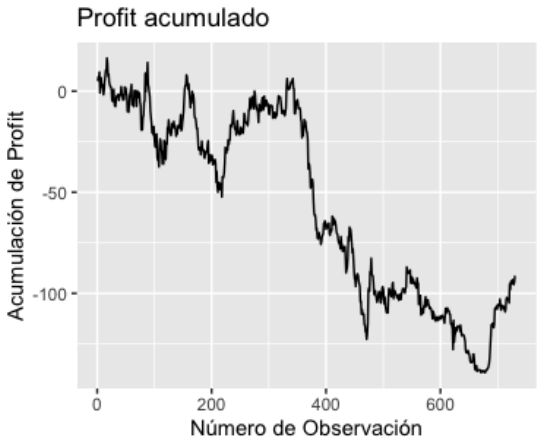


Gráfico 7: Rendimientos del estimador 3 (y) en el periodo de 2 años

Los resultados obtenidos bajo este enfoque se detallan en la Tabla 1. El estimador logra un Accuracy del 0.45, un profit acumulado de -91 y un PL-ratio de 0.87.

En el Gráfico 7, se muestra que la tendencia del profit acumulado indica un decrecimiento sostenido.

Para evaluar la consistencia de este estimador en particular, se lleva a cabo un análisis a lo largo de un período de 8 años. La tendencia general se muestra alcista, aunque se identifica un breve período de disminución. Es importante señalar que este período de baja es considerado atípico, ya que coincide con un año de cambios estructurales excepcionales en el mercado. Más allá de este lapso inusual, se sostiene que el crecimiento del estimador es consistente.

4.4. Estimador 4

En la lógica del Estimador 4 en una primera etapa, se define una ventana inicial en la que se emplean modelos lineales, utilizando el método StepWise,

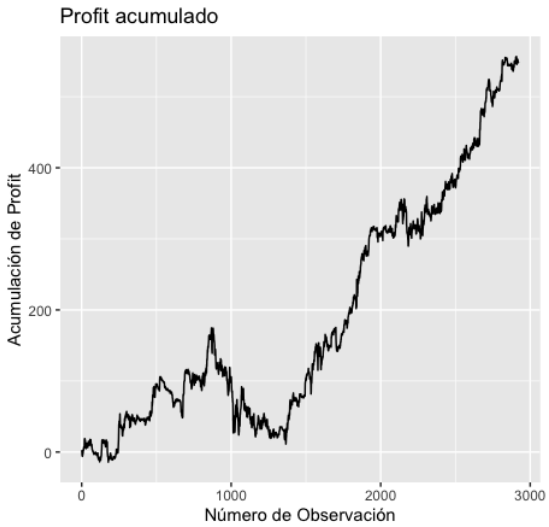


Gráfico 8: Rendimientos del estimador 2 Negative X en el periodo de 8 años

para identificar el modelo que mejor explica los $stock\_returns$.

Luego, se realiza un análisis univariado de cada una de las variables seleccionadas utilizando redes neuronales recurrentes. Esto permite obtener predicciones para el tiempo $t + 1$ para cada variable X.

Finalmente, se evalúa el modelo lineal previamente obtenido utilizando estas predicciones y se calcula la predicción del retorno simple en el momento $t + 1$.

Los resultados obtenidos bajo este enfoque se detallan en la Tabla 1. El estimador logra un Accuracy del 0.50, un profit acumulado de -50.7 y un PL-ratio de 0.94.

En el Gráfico 7, se muestra que la tendencia del profit acumulado si bien es decreciente, tiende a oscilar con subidas y bajadas.

5. Conclusiones

Durante la mayoría del análisis, se consolidó la noción de que los retornos simples siguen un patrón de paseo aleatorio. La mayoría de los estimadores mostraron una precisión del 0.5 o con diversos patrones. No obstante, nuestra investigación logró identificar un patrón que desafía este comportamiento aleatorio, evidenciado a través del Estimador 1 y el Estimador 2 (Negative X), los cuales exhiben una tendencia de crecimiento sostenida.

Este hallazgo respalda nuestra hipótesis inicial de que es factible descomponer el precio del Bitcoin y escapar del patrón de paseo aleatorio al analizar variables específicas con información privilegiada proveniente de la cadena de bloques. La coherencia en el crecimiento de las ganancias de estos estimadores sugiere que el modelo ha identificado patrones subyacentes en los datos, los

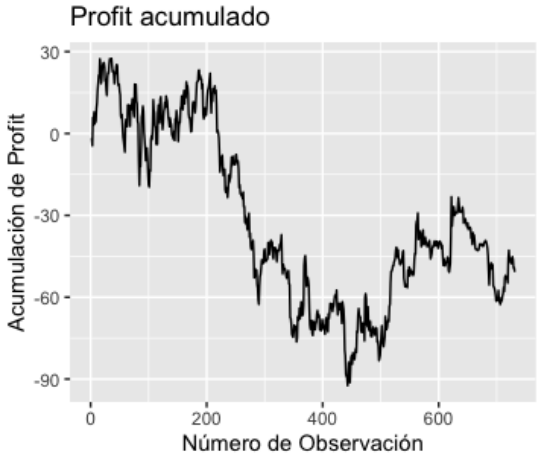


Gráfico 9: Rendimientos del estimador 4 en el periodo de 2 años

cuales son replicables y explotables.

A pesar de que todos los estimadores se evaluaron durante un período de 2 años, se realizo un análisis exhaustivo de 8 años en el Estimador 2 (Negative X), que es el que obtuvo mejor rendimiento y en este proceso se confirmó la generalización de este comportamiento, superando esta prueba de manera exitosa, demostrando la consistencia del estimador.

Estos resultados indican la posibilidad de establecer una estructura rentable basada en este estimador en particular. Además, abren una nueva línea de investigación para profundizar y desarrollar estrategias en busca de otros patrones y mayores rendimientos en el mercado de Bitcoin.

6. Pasos a seguir

Durante este estudio, se logró profundizar en las líneas de trabajo planteadas. Sin embargo, durante la fase de exploración e investigación, surgieron indicios de otras áreas prometedoras que podrían conducir a mejoras en los rendimientos de los estimadores o descubrir nuevos patrones. Por ejemplo, se identificaron posibilidades como el uso de Modelos de Markov Ocultos para detectar regímenes en las variables. Además, la aplicación de Análisis de Componentes Principales (PCA) para resumir la información de las variables y técnicas más avanzadas de aprendizaje automático, especialmente aquellas relacionadas con el aprendizaje por refuerzo (reinforcement learning), podría ser valiosa, especialmente cuando se aplica a la toma de decisiones sobre estrategias.

Explorar estas áreas adicionales de investigación no solo podría enriquecer los resultados existentes, sino también abrir nuevas puertas para descubrir patrones más complejos y estrategias más efectivas.

Estas direcciones potenciales ofrecen oportunidades significativas para am-

pliar la comprensión y la aplicación de los métodos analíticos en el ámbito de las finanzas de cryptomonedas.

A. Anexo A: Datos Adicionales

Referencias

[1] A. Arratia. *Computational Finance: An Introductory Course with R.* Atlantis Press (Zeger Karssen), 2014.

Variables seleccionadas	Descripción
variable_1	Es la suma acumulada en un período de 90 días de los Çoin Days Destroyed"(CDD), que representa la cantidad de días de antigüedad de las monedas que se han gastado durante el último año.
variable_4	Mide edad promedio de monedas sin moverse.
variable_5	es la edad promedio (en días) de las salidas de transacciones gastadas.
variable_6	La cantidad total de monedas en direcciones de exchange
variable_8	Binary CDD se utiliza para minimizar el impacto de los movimientos en los intercambios, que no reflejan con precisión el comportamiento de los tenedores a largo plazo.
variable_9	El Índice de Volatilidad de Bitcoin (BVIN) es un índice de volatilidad implícita que también representa el valor justo de un contrato de variación de bitcoin.
variable_10	Coin Days Destroyed (CDD) para cualquier transacción dada se calcula tomando la cantidad de monedas en una transacción y multiplicándola por el número de días que han transcurrido desde que esas monedas fueron gastadas por última vez.
variable_11	Coin Years Destroyed (CYD) se define como la suma acumulada de Coin Days Destroyed (CDD) durante un período de 365 días consecutivos.
variable_13	Difficulty Ribbon Compression.
variable_15	El cambio de 30 días en el suministro almacenado en las billeteras de intercambio.
variable_16	El Múltiplo de la Proporción de Tarifas (FRM) se define como la relación entre los ingresos totales de los mineros (recompensas por bloques tarifas de transacción) y las tarifas de transacción.
variable_17	El Hash Ribbon indica que lo peor de la capitulación de los mineros ha terminado cuando el promedio móvil de 30 días de la tasa de hash cruza por encima del promedio móvil de 60 días.
variable_18	Conjunto de todas las franjas de antigüedad del suministro activo, también conocidas como 'olas de retención' (HODL waves).
variable_19	El porcentaje de nuevas monedas emitidas, dividido por el suministro actual.

Cuadro 2: Variables sin precisel seleccionadas y descripciones

Variables seleccionadas	Descripción
variable_20	La cantidad total de nuevas monedas añadidas al suministro actual, es decir, monedas recién acuñadas o nuevas monedas puestas en circulación en la red
variable_21	La "Liveliness"se define como la relación entre la suma de los Çoin Days Destroyedz la suma de todos los días de moneda jamás creados. La vitalidad aumenta a medida que los titulares a largo plazo liquidan posiciones y disminuye cuando acumulan para mantener.
variable_24	El número promedio estimado de hashes por segundo producidos por los mineros en la red.
variable_25	Median Spent Output Lifespan (MSOL) Es la mediana de la antigüedad (en días) de las salidas de transacciones gastadas.
variable_26	El número estimado actual de hashes necesarios para minar un bloque.
variable_27	La diferencia en el volumen que fluye hacia los intercambios y el volumen que fluye fuera de los intercambios.
variable_31	número de direcciones activas de Bitcoin.
variable_32	número de direcciones de Bitcoin con un saldo no nulo
variable_33 a variable_39	número de direcciones de Bitcoin con balances entre 0.01 y 10k.
variable_40	número de direcciones de Bitcoin
variable_41	número de Bitcoins minados
variable_42	número de nuevas direcciones
variable_43	número de billeteras que reciben transacciones
variable_44	número de billeteras que envian transacciones
variable_45	número de transferencias entre exchanches
variable_46	número de transferencias desde exchanches
variable_47	número de transferencias a exchanches
variable_48	porcentaje de bitcoins en exchanches
variable_49	El porcentaje de los ingresos de los mineros derivados de las tarifas
variable_50	El porcentaje de la oferta circulante que no ha tenido movimiento en al menos 2 años.
variable_51	El porcentaje de la oferta circulante que no ha tenido movimiento en al menos 3 años.
variable_52	El porcentaje de la oferta circulante que no ha tenido movimiento en al menos 5 años.
variable_54	Precio del Bitcoin.

Cuadro 3: Variables sin precio seleccionadas y descripciones

Variables seleccionadas	Descripción
variable_67 a 79	Lifespam.
variable_81 a 93	spent volumen
variable_96	.ªdjusted Coin Days Destroyed"simplemente divide ÇDD"por la oferta circulante (cantidad total de monedas emitidas).
variable_97	CYD Çoin Years Destroyed (CYD)"se define como la suma acumulada de 365 días de Çoin Days Destroyed (CDD)z muestra la cantidad de días de moneda que han sido destruidos en el último año. Esto indica el comportamiento de los poseedores a largo plazo.
variable_98	La Dormancia es el promedio de días destruidos por moneda transaccionada y se define como la relación entre los días de moneda destruidos y el volumen total de transferencia.
variable_99	Esta métrica desglosa el suministro total de Bitcoin por tipos de salidas de transacción. Los tipos de salidas de transacción, o tipos de txout en resumen, se determinan según el tipo de condiciones de script de Bitcoin que se utilizan para bloquear Bitcoin en la salida. Los tipos de salidas de transacción más comunes son: P2TR (Paga a Taproot): Los fondos están bloqueados utilizando un hash de 32 bytes que puede ser (1) una clave pública, (2) una combinación de múltiples claves públicas o (3) un hash de script. P2WPKH (Paga a la Huella de Clave Pública Testigo): La versión SegWit de P2PKH. P2WSH (Paga a la Huella de Hash de Script Testigo): La versión SegWit de P2SH. Este tipo tiene dos variantes. P2SH (Paga a la Huella de Hash de Script): Los fondos están bloqueados utilizando instrucciones de script de Bitcoin arbitrarias. P2PKH (Paga a la Huella de Clave Pública): El sucesor de P2PK, bloqueando fondos utilizando el hash de una clave pública. P2PK (Paga a Clave Pública): El primer tipo disponible, bloqueando fondos utilizando una clave pública. Otros: Sirve como un contenedor para todas las salidas cuyas instrucciones de script no coinciden con ninguno de los tipos de salida más comúnmente utilizados enumerados anteriormente.

Cuadro 4: Variables sin precio seleccionadas y descripciones.

Variables seleccionadas	**Descripción**
variable_100	El tamaño total de todos los bloques creados dentro del período de tiempo (en bytes).
variable_101 al 113	El porcentaje de la oferta circulante que no ha tenido movimiento durante al menos 1 año.
variable_114	La cantidad total de tarifas pagadas a los mineros.
variable_115	El volumen total de transferencia de Bitcoin entre exchanges (BTC de todos los exchanges a todos los exchanges).
variable_116	Volumen total de transferencia.
variable_117	Volumen total de transferencia desde intercambios.
variable_118	Volumen total de transferencia hacia intercambios.
variable_119	Volumen total de transferencia dentro de intercambios.

Cuadro 5: Variables sin precio seleccionadas y descripciones.

Variables seleccionadas	**Descripción**
variable_2	El SOPR es una métrica que intenta medir la rentabilidad de los inversores que han vendido sus bitcoins en comparación con cuando los adquirieron por última vez.
variabl_3	Thermocap se calcula como la suma acumulada de recompensas en bloque en USD pagadas a los mineros. Refleja el 'costo de producción' agregado de todos los BTC en circulación
variabl_7	Precio Balanceado
variabl_12	El valor de días destruidos múltiples compara el comportamiento del gasto a corto plazo con el promedio anual, como medio para detectar mercados sobrecalentados e infravalorados.
variable_14	adjusted-dormancy-flow.
variable_22	market-cap-to-thermocap.
variable_23	market-value-to-realized-value-ratio.
variable_27	net-realized-profit-loss.
variable_29	network-value-to-transactions-ratio.
variable_30	network-value-to-transactions-signal.
variable_53	price-drawdown-from-ath.
variable_55	puell-multiple.
variable_56	realized-cap-hodl-waves.
variable_57	realized-hodl-ratio.
variable_58	realized-loss-usd.
variable_59	realized-profit-loss-ratio.
variable_60	realized-profit-usd.
variable_61	realized-profits-to-value-rpv-ratio.
variable_62	relative-unrealized-loss.
variable_63	Bitcoin relative unrealized profit
variable_64	Bitcoin reserve risk
variable_65	Bitcoin seller exhaustion constant
variable_66	Bitcoin spent output age bands
variable_80	Bitcoin spent output profit ratio SOPR
variable_94	Bitcoin stablecoin supply ratio SSR oscillator
variable_95	Bitcoin stablecoin supply ratio SSR
variable_101	Bitcoin total supply in loss BTC
variable_102	Bitcoin total supply in profit BTC
variable_120	Bitcoin velocity

Cuadro 6: Variables con precio seleccionadas y descripciones