



LOS LIBERTADORES
FUNDACION UNIVERSITARIA

Análisis Predictivo de la crypto moneda Monero

Predictive Analysis of Monero Crypto Currency

Juan David Aguilar

Mario Gregorio Saavedra

Resumen:

El mercado de las crypto monedas ha crecido en los últimos años siendo la más conocida Bitcoin, pero se han creado otras crypto monedas con las características de encriptar todo tipo de transacción como lo es Monero (XMR). Utilizando esta crypto moneda vamos a identificar primero si la serie presenta un comportamiento estocástico o determinístico utilizando el método de surrogate data testing y por último conociendo una aproximación a la estructura de la serie vamos a predecir diferentes intervalos de tiempo para corroborar si nuestro modelo identifica adecuadamente la tendencia de la serie.

Al realizar el método de surrogate data testing encontramos que los resultados obtenidos no fueron concluyentes utilizando diferentes estadísticos como Nolinear Prediction y Entropy complexity sobre la serie, por tal motivo no podemos afirmar que la serie se comporta como un proceso no lineal debido a este resultado realizamos una estimación lineal estocástica de la serie dividida en dos periodos pues se encontró que la estructura de la serie fue afectada por la pandemia covid-19. Utilizando el último periodo comprendido desde 2020-03-12 hasta 2020-10-18 se obtuvo un modelo auto regresivo de orden uno AR(1), el cual explicaba adecuadamente la serie y cumpliendo con los supuestos del modelo además se realizaron predicciones para diferentes días desde 2020-10-20 comparando con los datos reales del mercado, el resultado obtenido fue la correcta identificación de la tendencia de la serie, la cual se encontraba en los intervalos de confianza considerados.

Palabras Clave: Series De Tiempo, Auto Regresivo, Surrogate Data Testing, Crypto Monedas, Forecast.



Abstract:

The Crypto market has grown in the last years with different types of currencies, some of them are transaction public and other encrypted like Monero (XMR). Using this crypto currency, we want to identify if his behavior is described by a linear stochastic process or a deterministic no linear process using surrogate data testing and based on that create a model to predict the behavior of the series.

The result obtained by the surrogate data testing method using Nolinear Prediction and Entropy complexity statistics were not conclusive, this shows us that an approximation of the underling process is not deterministic no-linear process based on that we model the series with a linear stochastic process and we found the necessity to split the series in two parts before COVID-19 and after because this event affected drastically the market behavior. The final model is an Auto Regressive with one lag using data from 2020-03-12 to 2020-10-18 being able to correctly identify the tendency of the Monero series in different time points.

Keywords: Time Series, Auto Regressive, Surrogate Data Testing, Crypto Currencies, Forecast.



Introducción

En la actualidad se puede hacer trading con crypto monedas en diferentes plataformas como lo es Kraken.com que emula un mercado de capitales, el cual permite la venta y compra de activos financieros, sin estos mercados se incurriría en altos costos financieros al buscar un comprador a un activo además añade instrumentos de garantía para ambas partes en una transacción (Aragonés & Mascareñas, 1994), una característica importante de un mercado de capitales es la propiedad de eficiencia; es decir, la información del pasado no permite predecir el comportamiento futuro del mercado (Fama, 1970), este concepto permite que existan diferentes actores financieros como especuladores los cuales anticipan los cambios de precios para obtener posiciones que generen ganancias a corto plazo.

En el mercado de las crypto monedas se ha encontrado que la propiedad de eficiencia no se cumple en ventanas de tiempo bastante grandes (Kristoufek, 2018) (Kyriazis, 2019), lo anterior plantea la posibilidad de utilizar métodos para pronosticar el comportamiento de diferentes crypto monedas y en los últimos años ha existido un gran interés en predecir este comportamiento (Garg & Anupriya, 2018) (Bush & Choi, 2019) (Pichl, Eom, Scalas, & Kaizoji, 2020) (Akyildirim, Goncu, & Sensoy, 2020) (Wu, Lu, Ma, & Lu, 2018) (Kumar, 2019) pero se suelen enfocarse en las crypto monedas más famosas como Bitcoin o Ethereum.

Muchos de los modelos generados para predecir crypto monedas se basan en modelos no lineales como redes neuronales (Livieris, Pintelas, Stavroyiannis, & Pintelas, 2020) (Nasekin & Chen, 2020) (Alessandretti, ElBahrawy, Aiello, & Baronchelli, 2018) sin antes comprobar que realmente esta serie posee comportamiento no lineal estocástico o no tienen en cuenta las transformaciones necesarias para que el proceso sea estacionario (Guerrero, 2013). Lo anterior puede ocasionar que modelos como redes neuronales tengan un sobreajuste al ruido inherente que existe en los datos ya que no poseen forma de diferenciar los componentes de la serie (Goodfellow, Bengio, & Courville, 2016). Por este motivo es necesario comprobar por medio de pruebas estadísticas si la serie posee una estructura lineal estocástica o es no lineal determinística.



Otra característica importante de los métodos utilizados se halla en el formato de los datos, muchos de estos utilizan “closed price” (Kumar, 2019) (Garg & Anupriya, 2018); es decir, el precio al cierre del día. El cual no tiene aplicación directa pues un inversor no espera al final del día para invertir en una crypto moneda lo hace durante el mismo por lo que un modelo utilizando este tipo de datos no debería predecir un valor en concreto sino un intervalo de confianza para que el inversor conozca al día siguiente el rango donde la moneda se va a encontrar, este último enfoque es el que se realiza en este trabajo.

Las series de tiempo se basan en un proceso generador, el cual puede ser estocástico o determinístico, este último incluso podría tener propiedades caóticas las cuales podrían simular características estocásticas (Huffaker, Bittelli, & Rosa, 2018). Algunos autores afirman que las crypto monedas poseen comportamientos caóticos no lineales (Lahmiri & Bekiros, 2019), la anterior afirmación es difícil de sustentar pues las propiedades caóticas solo convergen cuando se tiende al infinito y los datos no poseen ruido (Hilborn, 2006) algo que no ocurre con la mayoría de series de tiempo por lo que en nuestro caso nos vamos a enfocar en utilizar estadísticos como exponente de Lyapunov, NoLinear Prediction method y Permutation Entropy para generar pruebas no paramétricas con el objetivo de identificar si la serie posee comportamiento lineal estocástico o no lineal determinístico (Huffaker, Bittelli, & Rosa, 2018).

En este contexto, generar un modelo para predecir la tendencia de una crypto moneda requiere primero despejar la incógnita si una aproximación al proceso generador es lineal estocástico, lineal determinístico o no lineal mediante una prueba no paramétrica. El anterior paso nos permite descartar que la variabilidad de la serie sea caótica o estocástica, y es de suma importancia pues nos va a permitir restringir el tipo de modelos a utilizar. Por último, según el modelo seleccionado se deberá generar intervalos de confianza para que el inversor conozca el rango donde la crypto moneda va a estar en futuros días ya que por la característica de los datos obtenidos (closed price) no es útil generar valores exactos.



Referentes Teóricos

BlockChain, es una plataforma transaccional descentralizada que permite a cada nodo de la red verificar la autenticidad de una transacción; para ello cada nodo almacena de manera codificada una cadena de bloques, la cual contiene información de la transacción, bloque anterior y posterior, esto permite realizar un consenso para verificar la validez de una transacción (HOUBEN & SNYERS, 2018) (TAPSCOTT & TAPSCOTT, 2016), eliminando la necesidad de terceros como instituciones financieras, entidades gubernamentales, o empresas que prestan servicios de transferencia de servicios o activos financieros. Se pueden encontrar varias aplicaciones que utilizan esta tecnología para registrar transacciones como lo son registro de propiedades inmobiliarias, plataformas de CarSharing, Cloud Storage entre otras.

Las Crypto monedas, son monedas virtuales; es decir, no toman una forma física en papel u otro material, las cuales poseen la característica de necesitar una llave para poder leer el contenido de esta y se pueden clasificar de diferentes maneras una de ellas es basado en el tipo de relación con el mercado real, se pueden destacar tres tipos:

- I. Monedas Virtuales que solo pueden ser utilizadas en sistemas cerrados, se suelen utilizar en plataforma de video juegos (e.g Fifa points, World of Warcraft Coins)
- II. Monedas Virtuales que están unilateralmente unidas al mercado real: una tasa de conversión existe para la compra de una moneda (con dinero tradicional) con la cual se puede comprar bienes o servicios (e.g Facebook Credits)
- III. Monedas Virtuales que están bilateralmente unidas al mercado real: existe una tasa de conversión para comprar y vender utilizando dinero tradicional; la cual permite comprar bienes o servicios. (HOUBEN & SNYERS, 2018)



La crypto moneda más conocida es el Bitcoin, la cual encaja en la tercera clasificación y es una moneda que funciona utilizando la plataforma de BlockChain; es decir, no tiene una entidad central que controle o manipule el precio ya que su valor es dado por la oferta y demanda de este, donde cada transacción es validada por la red.

También existen cientos de crypto monedas que se les suele llamar Altcoins, o monedas alternativas al Bitcoin (Virtual Currencies Key Definitions and Potential AML/CFT Risks, 2014), las cuales se pueden clasificar en dos:

- I. Altcoins que están creadas con el protocolo de comunicación original de Bitcoin añadiendo diferentes características, un ejemplo de ellas es Litecoin.
- II. Altcoins que no utilizan el protocolo de comunicación de Bitcoin, creando su propio protocolo y versión de Blockchain. Siendo las más conocidas Ethereum y Ripple. (HOUBEN & SNYERS, 2018).

Monero (XMR) es una Altcoin que posee la característica de privacidad, es decir, todo tipo de transacción es encriptada siendo imposible conocer el monto y el destinatario (Identificador) de esta a diferencia de otras crypto monedas como Bitcoin o Ethereum, donde es posible conocer este tipo de información. Cabe destacar que no es la única crypto moneda con estas características existen otras como Dash y Zcash.

Es importante destacar que el uso de Monero trae varias ventajas sobre crypto monedas “publicas”; por ejemplo, en Bitcoin es posible conocer el monto, el vendedor y el destinatario (IP) de la transacción por lo que existe la posibilidad de conocer si un gran comprador va a alterar el mercado comprando o vendiendo muchas crypto monedas generando oportunidades de arbitraje y haciendo que el mercado no sea eficiente. Otra ventaja es la protección ante ataques, como es posible rastrear los movimientos que ha tenido cierto usuario (IP) este puede ser víctima de ataques cibernéticos si se sabe que posee un portafolio muy grande en crypto monedas. Otra característica de esta crypto moneda es que permite ser completamente fungible; es decir no puede ser bloqueada por una entidad debido a transacciones ilegales (HOUBEN & SNYERS, 2018).



Metodología

El primer paso del proceso es utilizar el método de surrogate data testing para rechazar la hipótesis nula de un proceso estocástico lineal utilizando diferentes estadísticos como Máximo exponente de Lyapunov, Nolinear Prediction y Permutation Entropy. Para hacerlo vamos a generar 299 series mediante el método AAFT que provengan de un proceso lineal estocástico conservando diferentes propiedades para luego crear un test de cola superior o inferior los cuales nos van a permitir rechazar o aceptar la hipótesis nula.

El segundo paso consiste en modelar un proceso univariado lineal estocástico (ARIMA), lo anterior implica que la serie debe ser estacionaria, para ello estabilizamos primero la varianza de la serie y luego la media para evitar cambiar la estructura de los datos además realizamos dos pruebas de raíz unitaria para comprobar el supuesto de estacionariedad. El siguiente paso es graficar las funciones de auto correlación simple y parcial para conocer el orden de los parámetros AR(p) y MA(q) del modelo comprobando que los residuos del modelo no posean ninguna auto correlación no modelada.

Por último, utilizando diferentes criterios seleccionamos el modelo que mejor explique el comportamiento de los datos y realizamos un back testing en diferentes intervalos de tiempo para las fechas comprendidas entre 2020-10-23 hasta el 2020-11-20 mostrando dos intervalos de confianza al 50% y 95% verificando que los datos caen entre estos rangos.

Spectrum Analysis

El análisis del espectro nos permite identificar todas las frecuencias que componen una serie y sus componentes periódicos, para ello calculamos la transformada rápida de Fourier (Fast Fourier Transform) en la serie y calculamos el periodograma.

$$\left|d\left(\frac{j}{n}\right)\right|^2 = \frac{1}{n} \left(\sum_{t=1}^n x_t \cos\left(\frac{2\pi t j}{n}\right) \right)^2 + \frac{1}{n} \left(\sum_{t=1}^n x_t \sin\left(\frac{2\pi t j}{n}\right) \right)^2$$



La anterior formula nos halla el periodograma de una serie, la versión escalado simplemente aplica una normalización como se muestra a continuación.

$$P\left(\frac{j}{n}\right) = \frac{4}{n} \left| d\left(\frac{j}{n}\right) \right|^2$$

(Shumway & Stoffer, 2016) menciona que un periodograma escalado es “la varianza muestral de cada componente de la frecuencia” donde valores grandes de $P\left(\frac{j}{n}\right)$ indican que esa frecuencia es predominante en la serie y valores pequeños pueden ser atribuidos a ruido.

Cuando aplicamos el periodograma escalado en la serie de Monero encontramos que la serie no posee ningún componente periódico que se puede destacar como se observa en la Figura 1. Además presenta un comportamiento muy parecido al generado por un proceso auto regresivo AR(1) (Shumway & Stoffer, 2016).

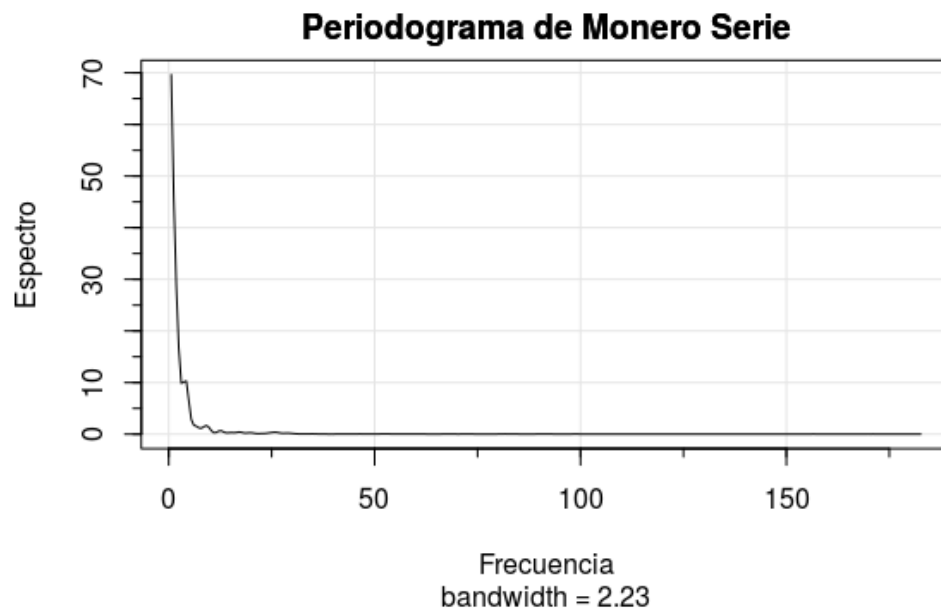


Figura 1: Periodograma de la serie Monero (XMR) suavizando el resultado utilizando el kernel de modified Daniel

Lo anterior nos da evidencia que la serie es estocástica en su mayoría y no posee componentes periódicos por lo que puede ser representado como un proceso lineal estocástico para comprobar la anterior hipótesis vamos a generar dos pruebas no paramétricas utilizando el método de surrogate data analysis.



Surrogate Data Analysis

Es un método que consta de cuatro partes. la primera consiste en plantear una hipótesis nula en nuestro caso se define como el proceso generador es lineal estocástico, el segundo paso es generar series (surrogates) basados en la hipótesis nula; por ejemplo, si nuestra hipótesis nula se basa en un proceso auto regresivo con latencia de orden uno generamos varias series basados en este parámetro. En este trabajo utilizamos el método de Amplitud Adjusted Fourier Transform (AAFT) para generar series de prueba las cuales preservan la distribución de probabilidad y la Power Spectrum de la serie, en específico utilizaremos el paquete “season” para generar las series de prueba (Barnett & Dobson, 2010).

El tercer paso consiste en estimar estadísticos de las series generadas y por último aplicamos pruebas de ranqueo no paramétricos (rank order no parametric test) para aceptar o rechazar la hipótesis nula. La anterior metodología se basa en el trabajo de (Huffaker, Bittelli, & Rosa, 2018).

Como ya conocemos la hipótesis nula y el proceso que genera las series de prueba (AAFT) vamos a generar los estadísticos para la serie, para esto primero debemos ver si es posible calcular los siguientes valores máximo exponente de lyapunov, Nolinear Prediction y Permutation Entropy sobre la serie.

Máximo Exponente de Lyapunov

El exponente de lyapunov mide la divergencia o convergencia de trayectorias las cuales inicialmente parten de una separación d_0 en el momento t_0 y su separación a tiempo t esta definida por

$$d(t) = d_0 e^{\lambda t}$$

Donde el parámetro λ es el exponente de Lyapunov para la trayectoria, si el valor de λ es positivo, entonces se puede decir que la serie es caótica (Hilborn, 2006). Cabe destacar que la anterior propiedad parte del supuesto en que no poseemos ruido en la serie y poseemos muchos datos; es decir, en el límite se cumple esta propiedad. Por este motivo en análisis de

series de tiempo donde el ruido es inherente y no se tienen muchos registros se suele utilizar este estadístico solo para discriminar si la serie es lineal o no.

Para calcular el máximo exponente de Lyapunov utilizamos la función “lyap_k” del paquete “tseriesChaos” la cual nos genera una gráfica con la evolución de la separación logarítmica vs el tiempo transcurrido como se muestra en la Figura 2.

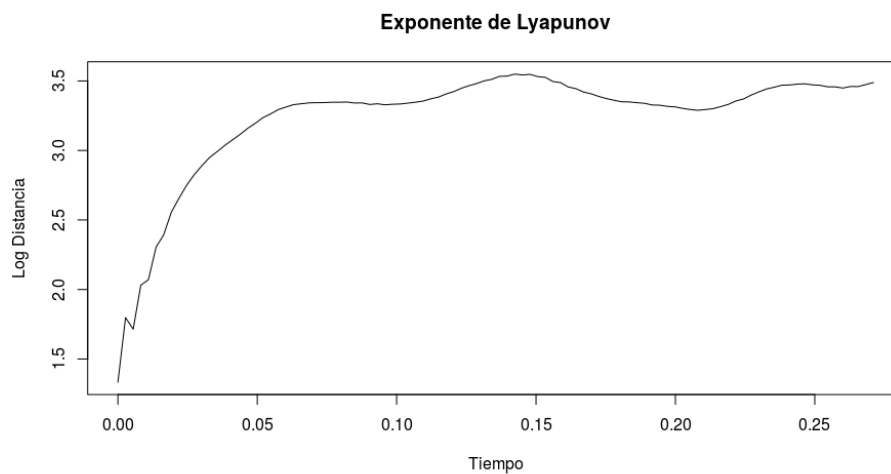


Figura 2: Gráfica de la evolución de la separación logarítmica vs tiempo para hallar el exponente de Lyapunov

Para calcular el máximo exponente de Lyapunov solo debemos generar una regresión lineal en el intervalo de tiempo donde exista una pendiente lineal, en la Figura 2 solo se puede observar una región muy pequeña con esta característica de 0.035 a 0.052 en el eje temporal por lo que obtenemos un estimado equivalente a 14.61 (Máximo exponente de Lyapunov). Cabe destacar que este valor estimado no es fiable pues la región donde es obtenido es muy pequeña por lo que no lo vamos a utilizar como estadístico para el análisis.



Nonlinear Prediction

Este método se basa en construir el espacio de fase mediante time delay embedding para luego crear ventanas de entrenamiento y de evaluación las cuales van a servir para medir la generalización de la serie. Mediante la métrica modified Nash-Sutcliffe (Nash & Sutcliffe, 1970) podemos medir la precisión de los resultados. Utilizaremos las funciones creadas por Huffaker y Bittelli para realizar este proceso (Huffaker, Bittelli, & Rosa, 2018).

Cuando aplicamos Nonlinear Prediction en los datos de Monero nos genera un valor de 0.92 para la métrica modified Nash-Sutcliffe, la cual indica que el proceso creado es muy preciso pues el coeficiente máximo que se puede alcanzar es 1 por lo que podemos utilizar este estadístico para las pruebas no paramétricas.

Permutation Entropy

Es un proceso que se suele utilizar como estadístico descriptivo en surrogate data analysis pues toma en cuenta que los datos son finitos y poseen ruido. Como se basa en la entropía su resultado es similar; es decir, un valor cercano a cero indica que el modelo se puede predecir perfectamente con los datos del pasado o no existe variabilidad en la serie y alcanza su valor máximo cuando los valores de la serie son independientes y uniformemente distribuidos (Huffaker, Bittelli, & Rosa, 2018). Se utilizó la función “entropyHeuristic” de la librería “pbc” para calcular la métrica Permutation Entropy sobre la serie, en donde obtuvimos un valor de 0.87 el cual vamos a utilizar en las pruebas no paramétricas.

Pruebas no paramétricas

Como se mencionó vamos a utilizar pruebas no paramétricas para aceptar o rechazar la hipótesis nula de un proceso lineal estocástico para ello vamos a definir un nivel de significancia del 95% ($\alpha = 0.05$) y el número de series de prueba para dos colas se calcula utilizando (Huffaker, Bittelli, & Rosa, 2018)

$$S = \left(\frac{2k}{\alpha}\right) - 1$$



En el caso de solo una cola utilizamos

$$S = \left(\frac{k}{\alpha}\right) - 1$$

Basados en lo anterior utilizamos una prueba de cola superior para el estadístico de Nonlinear Prediction, por lo cual rechazamos la hipótesis nula si el estadístico para la serie Monero predice con mayor precisión que las k máximos valores obtenidos del estadístico en las series de prueba.

Utilizamos una prueba de cola inferior para el estadístico de Permutation Entropy y rechazamos la hipótesis nula cuando el estadístico de la serie Monero es menor que el encontrado por las k mínimos valores obtenidos del estadístico en las series de prueba.

En nuestro caso generamos 299 series de prueba para realizar las pruebas no paramétricas y utilizamos un $k = 15$, obteniendo los siguientes resultados

	Monero	Surrogate (Low)	Surrogate (High)	H0
Nonlinear Prediction	0.92		0.88	Rechaza
Permutation Entropy	0.87	0.89		Rechaza

Tabla 1: Resultados de las pruebas no paramétricas utilizando el método de surrogate data analysis.

El resultado que obtuvimos fue rechazar la hipótesis nula de que el proceso es lineal estocástico en ambas pruebas; vemos que en la prueba Permutation Entropy rechazamos la hipótesis por solo dos puntos ante lo anterior no podemos afirmar contundentemente que el proceso sea no lineal pues el ruido que posee la serie puede generar resultados erróneos y al no existir una tendencia clara en la serie la cual nos permita separar dicho componente concluimos que solo rechazamos parcialmente la hipótesis nula.

Análisis Lineal Estocástico

Para poder crear un modelo lineal estocástico se necesita que la serie cumpla con diferentes supuestos uno de ellos radica en que la media y varianza sean constante y no dependan del tiempo, el segundo supuesto se basa en que la función de auto covarianza depende en s y t a través de la diferencia $|s - t|$. Si cumple con estos supuestos se denomina que la serie es débilmente estacionaria (Shumway & Stoffer, 2016).

Al graficar la serie Monero (XMR) vemos que no cumple con los supuestos ya que posee media y varianzas no constantes en el tiempo como se muestra en la Figura 3.

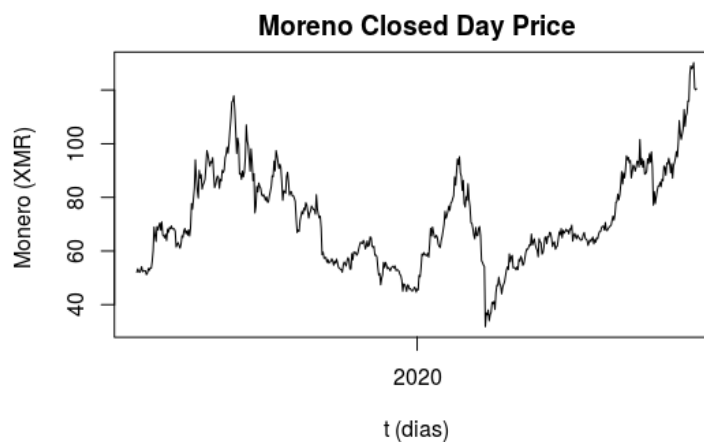


Figura 3: Gráfica de crypto Monero desde el 2019-03-15 hasta el 2020-10-18

Además, vemos que existen periodos donde el comportamiento es extraño por diferentes razones globales; un ejemplo de ello se destaca en el mes de marzo del 2020 donde se ve una clara disminución de valor probablemente debido a la pandemia registrada en ese periodo. Por lo cual vamos a dividir la serie en dos periodos, el primero periodo esta entre 2019-03-15 hasta 2020-03-12 y el segundo entre 2020-03-13 hasta 2020-10-18.



Estabilizar varianza

Sin importar en qué serie nos encontremos en ambos casos debemos estabilizar su varianza y media, por lo cual vamos a comenzar estabilizando la varianza aplicando las transformaciones de tukey (Hinkley, 1977)

$$y = \begin{cases} x^\gamma & \text{Si } \gamma > 0 \\ \log x & \text{Si } \gamma = 0 \\ -(x^\gamma) & \text{Si } \gamma < 0 \end{cases}$$

Verificando visualmente que la desviación estándar y la media sean constantes para periodos de una semana como se observa en la Figura 4 y 5 donde en ambos periodos se presenta esta característica.

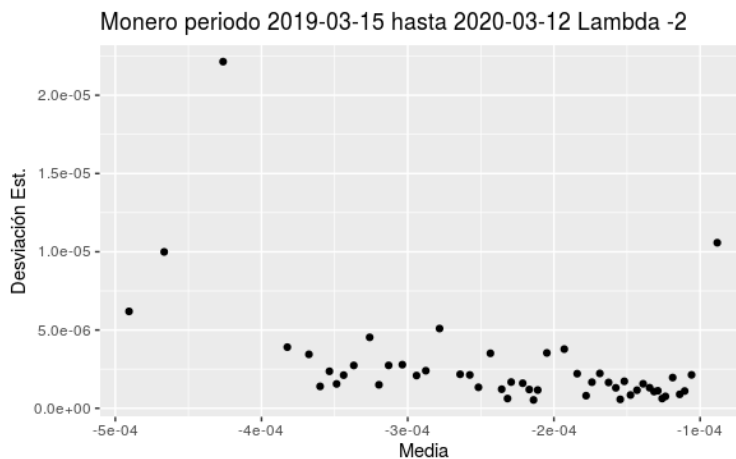


Figura 4: Gráfica del Monero para el periodo 2019-03-12 hasta el 2020-03-12 utilizando una transformación $-(x^{-2})$

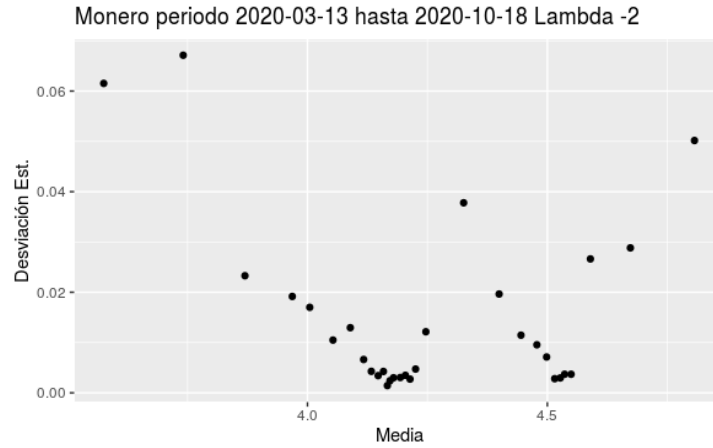


Figura 5: Gráfica del Monero para el periodo 2020-03-13 hasta el 2020-10-18 utilizando una transformación $\log(x)$

Estabilizar Media

Vamos a aplicar el método de diferencias para estabilizar la media en los dos periodos revisando visualmente si se cumple que la media sea constante. En las Figuras 6 y 7 se observa que aproximadamente la media es constante exceptuando el periodo de febrero y marzo del 2020 pues en esta fecha existió más volatilidad debido a la pandemia.

Pruebas de Raíz Unitaria

Basados en las pruebas graficas obtenidas donde encontramos que las series son estacionarias vamos a confirmar estos supuestos con dos pruebas de raíz unitaria, la primera es la prueba Aumentado de Dickey-Fuller y la prueba KPSS (Guerrero, 2013). Los resultados se muestran en la Tabla 2 donde se comprueba que las series son estacionarias utilizando un nivel de confianza del 0.05.

Prueba	H0	P valor	Conclusión
Dickey-Fuller	No estacionaria	0.01	Rechaza la H0
KPSS	Estacionaria	0.1	Acepta H0

Tabla 2: Resultados de pruebas de Raíz Unitaria.

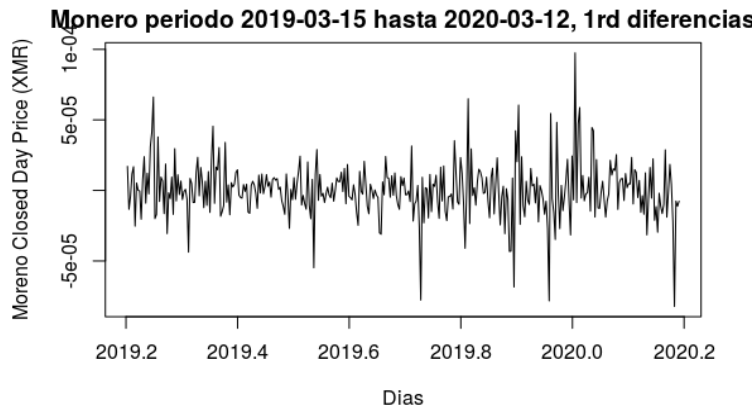


Figura 6: Gráfica de Monero para el periodo 2019-03-15 hasta el 2020-03-12 aplicando diferencias de orden uno.

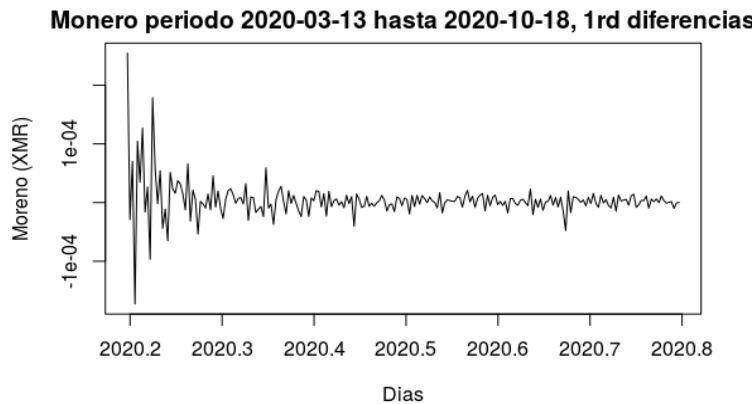


Figura 7: Gráfica de Monero para el periodo 2020-03-13 hasta el 2020-10-18 aplicando diferencias de orden uno.

Modelamiento ARIMA Primer Periodo

Comenzaremos modelando la serie para el primer periodo, para saber el orden auto regresivo (AR) y media móvil (MA) que posee vamos a graficar la función de auto correlación simple y parcial, la cual se muestra en la Figura 8, la cual tiene un comportamiento parecido al auto regresivo de orden uno.

Basados en lo anterior vamos a crear un modelo ARIMA de orden 1,1,0; es decir, auto regresivo de orden uno aplicando diferencias de orden uno. los resultados encontrados se



muestran en la Figura 9. Vemos que el modelo AR(1) cumple con los supuestos del modelo donde los residuos no muestran una tendencia y la prueba Ljung-Box no rechaza la hipótesis nula. Aunque existe el lag 20 donde se rechaza la hipótesis nula pero este caso representa el periodo asignado entre febrero y marzo del 2020 el cual representa un atípico. Por lo que podemos concluir que un modelo AR(1) representa de forma adecuada al periodo 2019-03-15 hasta 2020-03-12.

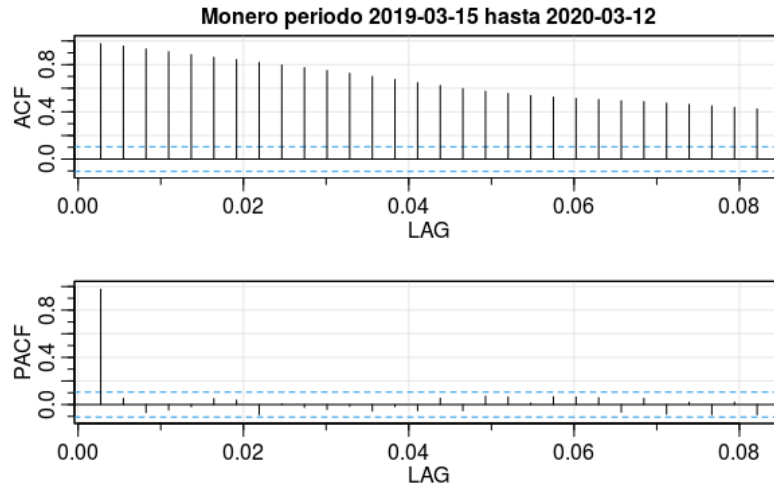


Figura 8: Gráficas de ACF y PACF para la serie de Monero.

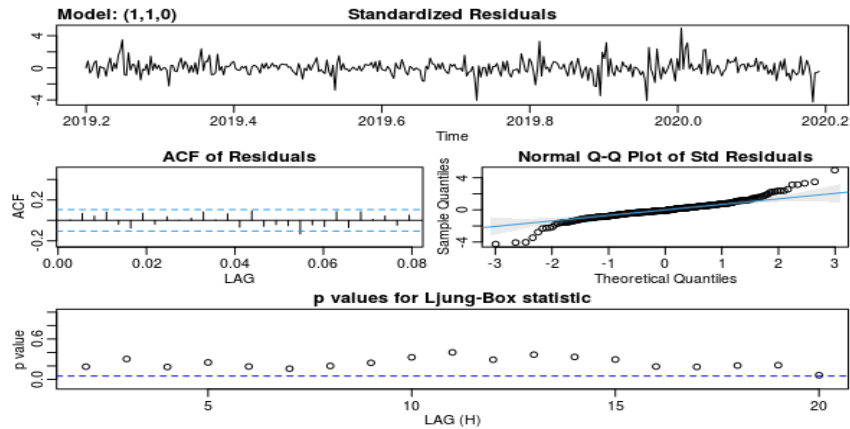


Figura 9: Resultados de aplicar un modelo ARIMA (1,1,0) sobre la serie en el periodo 2019-03-15 hasta el 2020-03-12.



Modelamiento ARIMA Segundo Periodo

En el segundo periodo comprendido desde 2020-03-12 hasta 2020-10-18 vamos a realizar el mismo proceso de generar las gráficas de la función de autocorrelación simple y parcial para descubrir el orden del proceso ARIMA. En la Figura 10 se observa que este orden puede ser un AR(1), MA(1) o un ARMA(1,1) por lo que vamos a generar esos tres modelos verificando los supuestos debidos.

La Figura 11 nos muestra los resultados del modelo AR(1), donde los residuos no muestran ninguna auto correlación entre ellos y la prueba Ljung-Box confirma que estos se comportan como ruido blanco por lo que podemos afirmar que este modelo representa bien los datos.

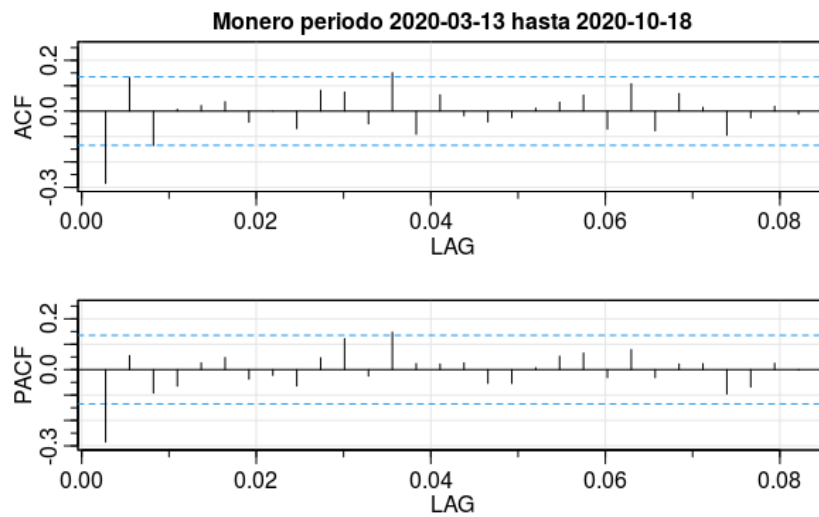


Figura 10: Gráficas de ACF y PACF para la serie en el periodo de 2020-03-12 hasta 2020-10-18.

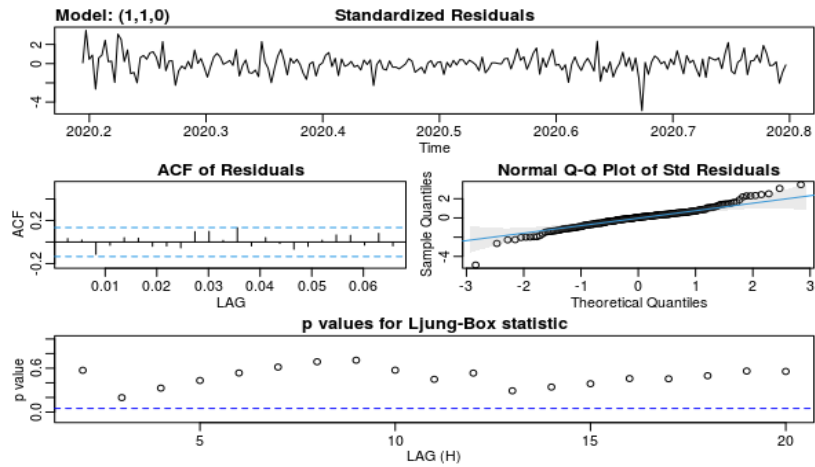


Figura 11: Resultados de aplicar un modelo ARIMA (1,1,0) sobre la serie en el periodo de 2020-03-12 hasta 2020-10-18.

En el caso del modelo MA(1), los resultados obtenidos nos muestran que en el lag número 3 de los residuos el test Ljung-Box esta muy cerca de ser rechazado por lo que no utilizamos este modelo, lo anterior se puede observar en la Figura 12.

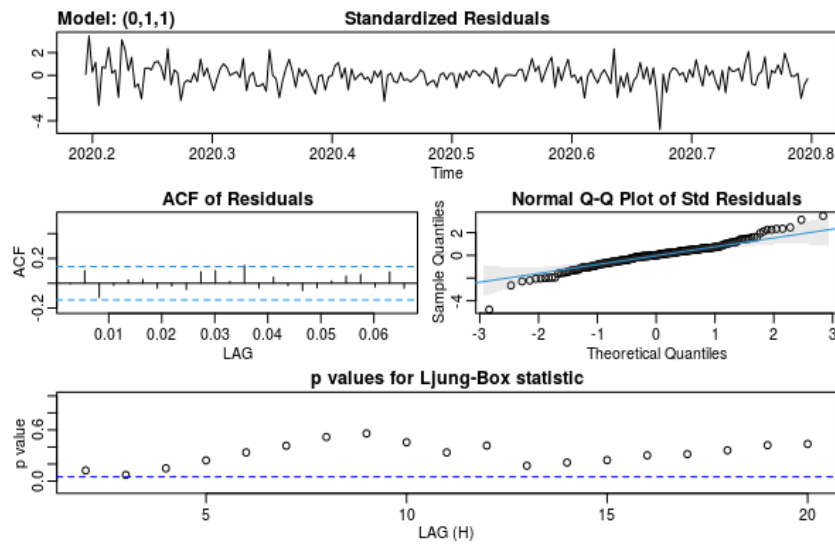


Figura 12: Resultados de aplicar un modelo ARIMA (0,1,1) sobre la serie en el periodo de 2020-03-12 hasta 2020-10-18.

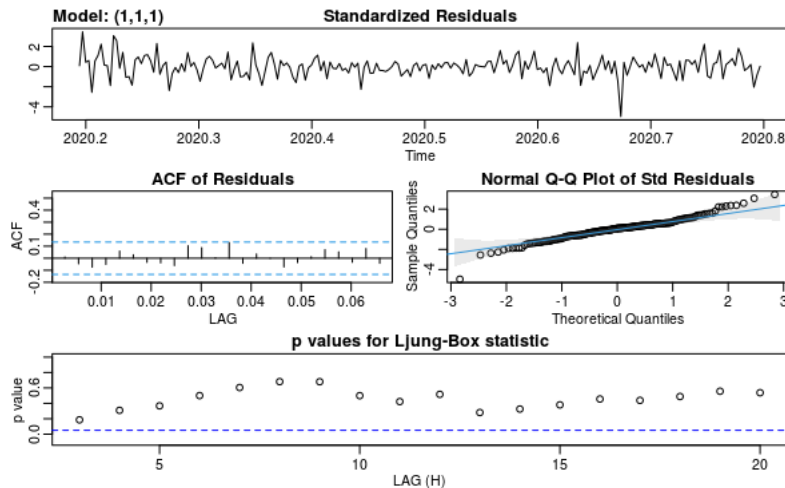


Figura 13: Resultados de aplicar un modelo ARIMA (1,1,1) sobre la serie en el periodo de 2020-03-12 hasta 2020-10-18.

Finalmente, la Figura 13 nos muestra un modelo ARIMA(1,1,1) en el cual se cumple todos los supuestos de no auto correlación en los residuos por lo que podemos utilizar este modelo.

Se obtuvieron dos modelos que cumplen con los supuestos AR(1) y ARIMA(1,1,1), basado en los criterios de log likelihood y AIC el modelo que explica mejor la serie es una auto regresión de orden uno auto regresivo de orden uno, $y_t = 0.0059 - 0.2997y_{t-1} + \epsilon$ y vamos a utilizar este modelo para predecir varios intervalos en el futuro.

Predicción de 10 días

Generamos seis predicciones en diferentes días utilizando un modelo AR(1) identificado en la sección anterior con los parámetros mostrados en la Tabla 3. El resultado que obtuvimos lo comparamos con los valores reales ocurridos en días posteriores al 2020-10-18 como se muestra en la Figura 14, se observa que el modelo identifica bien la tendencia de la crypto moneda Monero por diez días y los datos reales se encuentra en los intervalos de 95% y 50% confianza del modelo, lo anterior es de gran ayuda en algoritmos de trading.

Modelo AR(1)	Coefficientes	Desv. Estandar
AR(1)	-0.2997	0.0661
Media	0.0059	0.0020
Log Likelihood	399.18	
AIC	-792.35	

Tabla 3: Coeficientes del modelo AR(1) obtenidos utilizando el periodo de 2020-03-12 hasta 2020-10-18.

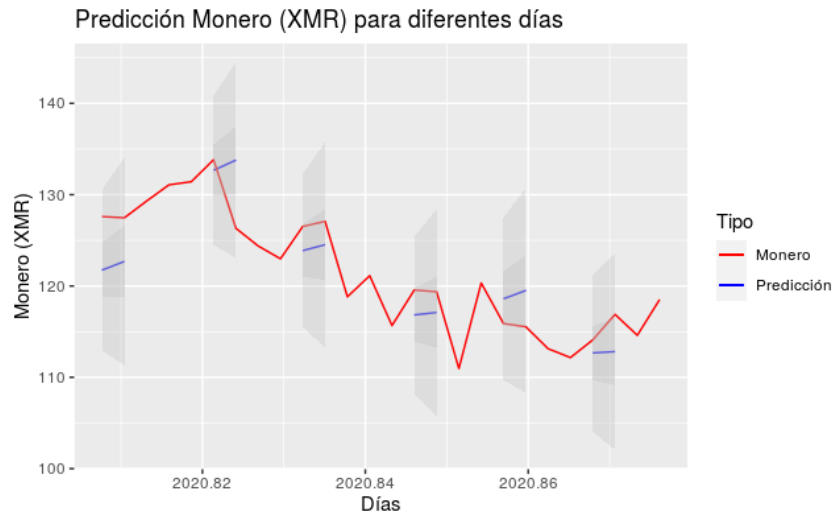


Figura 14: Comparación de modelo AR(1) con respecto a valores reales obtenidos después del 2020-10-18, además se muestra los intervalos de confianza del 50% y 95%.

Además, se muestran diferentes métricas obtenidas con la base de datos de test pero se debe tener en cuenta que estos valores no sirven para medir la calidad del modelo pues el objetivo del problema no es reducir el error del precio al cierre del día sino obtener intervalos de confianza donde nos dan guía hacia dónde va a estar el precio al día siguiente.

Predicción	MAE	MAPE	RMSE
1 Período 2020-10-23 (2 días)	5,317	0,042	8,345
2 Período 2020-10-30 (2 días)	8,481	0,068	5,567
3 Período 2020-11-06 (2 días)	9,033	0,079	5,107
4 Período 2020-11-10 (2 días)	3,846	0,034	3,9341
5 Período 2020-11-13 (2 días)	4,560	0,088	7,476
6 Período 2020-11-15 (2 días)	3,846	0,075	5,445

Tabla 4: Tabla de Métricas para los periodos pronosticados.



Conclusiones

Encontramos que el efecto generado por la pandemia en marzo cambio la estructura del comportamiento de la crypto moneda Monero por tal motivo dividimos la serie en dos periodos donde el segundo contenía los registros desde marzo 13 del 2020 hasta octubre 18 del 2020 y sobre este periodo modelamos un modelo de series de tiempo ARIMA(1,1,0) ó AR(1), el cual correctamente identifico la tendencia de la serie diez días después del 18 de octubre del 2020 como se observa en la Figura 14. El hecho de correctamente identificar la tendencia de una serie es de gran utilidad para algoritmos de trading o para minimizar el riesgo en un portafolio.



Bibliografía

- Agung, G. N. (2019). *Advanced Time Series Data Analysis*. Wiley.
- Akyildirim, E., Goncu, A., & Sensoy, A. (2020). Prediction of cryptocurrency returns using machine learning. *Annals of Operations Research*.
- Alessandretti, L., ElBahrawy, A., Aiello, L. M., & Baronchelli, A. (2018). Anticipating Cryptocurrency Prices Using Machine Learning. *Complexity*, 16-22.
- Aragonés, J. R., & Mascareñas, J. (1994). La eficiencia y el equilibrio en los Mercados de Capital. *Análisis Financiero*, 76-89.
- Barnett, A. G., & Dobson, A. J. (2010). *Analysing Seasonal Health Data*. Springer.
- Bush, R., & Choi, S. (2019). Forecasting Ethereum STORJ Token Prices: Comparative Analyses of Applied Bitcoin Models. *2019 International Conference on Data Mining Workshops (ICDMW)*, 216-223.
- Fama, E. (1970). Efficient Capital Markets: A Review of Theory and Empirical. *The Journal of Finance*, 383-417.
- Garg, S., & Anupriya. (2018). Autoregressive Integrated Moving Average Model based Prediction of Bitcoin Close Price. *International Conference on Smart Systems and Inventive Technology*.
- Goodfellow, I., Bengio, J., & Courville, A. (2016). *Deep Learning (Adaptive Computation and Machine Learning series)*. MIT Press.
- Guerrero, V. M. (2013). *Análisis Estadístico de Series de Tiempo Económicas*. Thomson.
- Hilborn, R. C. (2006). *Chaos and Nonlinear Dynamics*. Oxford.
- Hinkley, D. (1977). On Quick Choice of Power Transformation. *Applied Statistics*, 67-69.
- Huffaker, R., Bittelli, M., & Rosa, R. (2018). *No Linear Time Series Analysis with R*. Oxford.
- Kristoufek, L. (2018). On Bitcoin markets (in)efficiency and its evolution. *Physica A: Statistical Mechanics and its Applications*.
- Kumar, S. (2019). FORECASTING CRYPTOCURRENCY PRICES USING ARIMA AND NEURAL NETWORK: A COMPARATIVE STUDY. *The Journal of Prediction Markets*, 33-44.
- Kyriazis, N. (2019). A Survey on Efficiency and Profitable Trading Opportunities in Cryptocurrency Markets. *J. Risk Financial Manag.*, 12.
- Lahmiri, S., & Bekiros, S. (2019). CryptoCurrency Forecasting with deep learning chaotic neural networks. *Chaos, Solutions and Fractals*, 35-40.
- Livieris, I., Pintelas, E., Stavroyiannis, S., & Pintelas, P. (2020). Ensemble Deep Learning Models for Forecasting Cryptocurrency Time-Series. *Algorithms*, 121.



- Nasekin, S., & Chen, C. Y.-H. (2020). Deep learning-based cryptocurrency sentiment construction. *Digital Finance*, 39-67.
- Nash, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I — A discussion of principles. *Journal of Hydrology*, 282–290.
- Paolella, M. S. (2018). *Linear Models and Time-Series Analysis*. Wiley.
- Pichl, L., Eom, C., Scalas, E., & Kaizoji, T. (2020). Forecasting of Cryptocurrency Prices Using Machine Learning. In V. Derbentsev, A. Matviychuk, & V. N. Soloviev, *Advanced Studies of Financial Technologies and Cryptocurrency Markets* (pp. 211-231). Springer.
- Shumway, R. H., & Stoffer, D. S. (2016). *Time Series Analysis and Its Application with R examples*. Springer.
- TAPSCOTT, A., & TAPSCOTT, D. (2016). *La revolución blockchain*. Deusto.
- Wei, W. W. (2019). *Multivariate Time Series Analysis and Applications*. Willey.
- Wu, C., Lu, C., Ma, Y., & Lu, R. (2018). A New Forecasting Framework for Bitcoin Price with LSTM. *IEEE International Conference on Data Mining Workshops*, 168-175.