



Visión Gerencial
ISSN: 1317-8822
ISSN: 2477-9547
morillom@ula.ve
Universidad de los Andes
República Bolivariana de Venezuela

Mohali Marcano, Andrés Manuel; Palm Rojas, Francisco Javier
Efecto de indicadores económicos adelantados sobre la predicción de precios de criptomonedas
Visión Gerencial, núm. 1, 2022, Enero-Junio, pp. 53-69
Universidad de los Andes
Mérida, República Bolivariana de Venezuela

DOI: <https://doi.org/10.53766/VIGEREN/2021.21.01.03>

Disponible en: <https://www.redalyc.org/articulo.oa?id=465582602004>

- ▶ [Cómo citar el artículo](#)
- ▶ [Número completo](#)
- ▶ [Más información del artículo](#)
- ▶ [Página de la revista en redalyc.org](#)

redalyc.org

Sistema de Información Científica Redalyc
Red de revistas científicas de Acceso Abierto diamante
Infraestructura abierta no comercial propiedad de la academia

Efecto de indicadores económicos adelantados sobre la predicción de precios de criptomonedas

Andrés Manuel Mohali Marcano¹

Universidad de Los Andes. Mérida - Venezuela.
Correo electrónico: andresmohali1@gmail.com

Francisco Javier Palm Rojas²

Universidad de Los Andes. Mérida - Venezuela.
Correo electrónico: francisco.palm@gmail.com

INFORME DE INVESTIGACIÓN

Advanced economic indicators effect on cryptocurrency prices forecasting

RESUMEN

Este estudio analiza el desempeño de modelos complejos al incluir variables externas. Se compara el rendimiento de redes neuronales LSTM (Long-Short Term Memory), con y sin incluir las variables macroeconómicas del petróleo, el índice del dólar, el índice del euro y el cambio dólar/yuan junto con datos históricos de criptomonedas, para intentar mejorar las predicciones. Se proponen cuatro modelos para ambos casos, univariante y multivariante, comparando el mejor de cada caso, centrando los resultados en las variables en lugar de los modelos. Los mejores resultados LSTM se compararon entonces con el modelo ARIMA (Auto regresivo, Diferenciado y de Media Móvil). Usando los índices MASE (Media Absoluta Escalada del Error), MAPE (Media Absoluta Porcentual del Error) y el estadístico Diebold-Mariano. Se concluyó que había diferencias significativas para todas las series exceptuando al BitcoinCash. Siendo más precisos los modelos de redes neuronales univariantes. Asimismo, todos los modelos LSTM presentaron diferencias significativas con los modelos ARIMA. Siendo el LSTM el mejor de los casos, exceptuando al BitcoinCash. Se concluyó que las variables económicas consideradas no eran significativas para predecir los precios de las criptomonedas seleccionadas y que el ajuste de las redes LSTM debe hacerse de acuerdo con la complejidad del problema.

Palabras Clave: Criptomonedas, Redes Neuronales, LSTM, variables macroeconómicas, predicciones, aprendizaje automático.

Abstract

This study analyzes the performance of complex models when including external variables. It compares the performance of LSTM (Long-Short Term Memory) neural networks, with and without the macroeconomic variables of oil, dollar index, euro index and dollar/yuan exchange rate along with historical cryptocurrency data, in an attempt to improve predictions. Four models are proposed for both univariate and multivariate cases, comparing the best of each case, focusing the results on the variables rather than the models. The best LSTM results were then compared with the ARIMA (Auto Regressive, Differenced and Moving Average) model. Using the MASE (Mean Absolute Scaled Error), MAPE (Mean Absolute Percentage Error) indices and the Diebold-Mariano statistic. It was concluded that there were significant differences for all series except BitcoinCash. The univariate neural network models were more accurate. Likewise, all the LSTM models presented significant differences with the ARIMA models. The LSTM being the best of the cases, with the exception of BitcoinCash. It was concluded that the economic variables considered were not significant for predicting the prices of the cryptocurrencies considered and that the adjustment of the LSTM networks should be made according to the complexity of the problem.

Key words: Cryptocurrencies, Neural Networks, LSTM, macroeconomic variables, prediction, machine learning.

Recibido: 18-05-2021

Revisado: 20-07-2021

Aceptado: 30-08-2021

¹ Licenciado en Estadística de la Universidad de Los Andes ULA. Profesor e investigador del Departamento de Estadística, Facultad de Ciencias Económicas y Sociales, de la Universidad de Los Andes ULA, Venezuela. Líneas de Investigación: Desarrollo y aplicación de técnicas de aprendizaje automático

² Ing. En Sistemas de la Universidad de los Andes, ULA. M.Sc. Matemática Aplicada a la Ingeniería, ULA. Profesor investigador de la Universidad de Los Andes. Departamento de Estadística, Facultad de Ciencias Económicas y Sociales. Mérida - Venezuela. Líneas de investigación: Simulación y modelado de sistemas socioeconómicos, ambientales y urbanos. Geo estadística y análisis espacial.

Como citar este artículo - How to cite this article

Mohali, A. y Palm F. (2022). Efecto de los indicadores económicos adelantados sobre la predicción de precios de criptomonedas. *Revista Visión Gerencial*, 21(1), pp. 59-68. Recuperado de: <https://doi.org/10.53766/VIGEREN/2021.21.01.03>

1. Introducción

El 12 de enero de 2009 se lanza una tecnología que, hasta la actualidad, sigue evolucionando y cambiando nuestras vidas: El Bitcoin, una Criptomoneda basada en la tecnología blockchain con ventajas y características nunca antes vistas o al menos nunca antes aplicada al área económica. Una Criptomoneda es, en resumen, una moneda digitalizada que resuelve el problema del doble uso. Esto es, gastar la misma “moneda” más de una vez. Y tiene las interesantes particularidades de estar disponible a cualquiera con acceso a Internet y ser descentralizada, en la mayoría de los casos.

La posibilidad de cambios tan drásticos en el precio de estas monedas en tan poco tiempo, nos da la idea de la posibilidad de ganancia de mucho dinero en corto tiempo, así como sufrir pérdidas con la misma velocidad. En este sentido, queremos explicar la variabilidad de las Criptomonedas, con respecto a su valor, para aprovechar sus altos y bajos al tiempo de saber por qué y cuándo cambia su cotización. Optando por la perspectiva de que las criptomonedas están de una forma u otra respaldadas por las principales divisas con las que se negocian (dólar, euro). Es decir, gran parte de su valor está en la seguridad de poder transformarlas luego en estas divisas, podríamos pensar que existe una relación entre estas divisas, las economías de los países que las proveen y los mercados de criptomonedas.

En este artículo, basado en la tesis de grado del mismo nombre, con Andrés Mohali como tesista y Francisco Palm como tutor. Se estudia la idea de utilizar la información disponible de divisas fiduciarias y criptomonedas, para observar la eficiencia en la predicción del precio de las mismas.

2. Revisión de la literatura

2.1. Mercado de criptomonedas

Es posible verificar que las criptomonedas han servido más como mercados de valores de los cuales obtener ganancias que como dinero alternativo, estudiando el volumen de transacciones, donde criptomonedas como el Bitcoin pueden llegar a mover el equivalente a la capitalización total de su mercado en 7 días solo en casas de cambio. Aun cuando existen más de 3000 criptomonedas contando únicamente las registradas en Coinmarketcap, existen criptomonedas que predominan enormemente en el mercado, por lo que ellas capturan gran parte de la variabilidad. Así, vemos que tomando el Bitcoin, Ethereum, Bitcoin Cash, Litecoin y XRP se capta aproximadamente el 77% de la capitalización total del mercado de criptomonedas para el 24/02/2019.

2.2. Mercado de divisas

La mayoría de las cotizaciones y también los cambios en los operadores, ofrecen cambio solo con ciertas divisas (o en su defecto transformaciones de estas), principalmente Euro y Dólar. Esto puede hacernos pensar que, en realidad, estas monedas son dependientes de las divisas principales para comercializarlas y de las economías de los países que las proveen (Antoniadis, I. Et al. 2018). En pocas palabras, si las criptomonedas no pudieran cambiarse a dólares o euros, sería difícil que la gente las aceptara, es decir, perderían su valor.

Para medir las variaciones en la fortaleza de las principales economías, y de sus divisas utilizaremos las siguientes variables: Índice del dólar (ICE Futures U.S. 2015), Índice del euro, cambio dólar/yuan, y los precios del petróleo de la OPEP. Los índices del Dólar y del Euro se calculan de la forma:

$$\text{USDx} = 50.14348112 \times \text{EURUSD}^{-0.576} \times \text{USDJPY}^{0.136} \times \text{GBPUSD}^{-0.119} \times \text{USDCAD}^{0.091} \times \text{USDSEK}^{0.042} \times \text{USDCHF}^{0.036}$$

$$\text{EUR}_I = 51.58070891251366 \times \text{EURUSD}^{0.25} \times \text{EURGBP}^{0.25} \times \text{EURJPY}^{-0.25} \times \text{EURCHF}^{0.25}$$

Estas miden el valor de la moneda objetivo con el de una canasta de monedas extranjeras. Debido a que, ni en el índice del dólar ni del euro se contempla el cambio con el yuan, y entendiendo que la economía china actualmente lucha por ser la primera del mundo se contempla el cambio de yuanes contra el dólar.

2.3. Modelos de series de tiempo y de redes neuronales para modelar el mercado de las criptomonedas

Sabemos que modelos como ARIMA han tenido hasta 90% de precisión en periodos cortos de hasta 10 días (Roy et al, 2018). Otros resultados muestran que debido a la alta volatilidad del Bitcoin el modelo ARIMA acumula muchos errores, por lo que sirve principalmente para hacer predicciones utilizando sub-periodos de comportamiento estable, dando como resultado lógico predicciones únicamente a corto plazo (Azari, 2018).

Debido a que los modelos de redes neuronales se entrenan para una serie de datos específicos, lo que les quita generalidad, pero los especializa en el problema particular, existía la necesidad de saber cuál de estas técnicas presentaba mejores resultados a la hora de la estimación. Se ha demostrado que las redes neuronales LSTM generan mejores resultados que el modelo ARIMA (Karakoyun & Cibikdiken, 2018).

2.4. Teoría básica de redes neuronales y redes LSTM

Las redes neuronales pueden visualizarse como una serie de nodos interconectados. Estos son: Nodos de entrada, nodos ocultos y nodos de salida. La forma en que la red

“aprende” es a través de la minimización de una función de error, usando por lo general descenso de gradiente. La minimización de esta función de error implica el reajuste de los pesos de todas las neuronas, capa por capa, de atrás hacia adelante, debido a que la comparación se hace con el valor resultante de la red. A este proceso se le llama *back propagation* ya que va ajustando los pesos de la red de la capa de salida hacia la capa de entrada (Michael, A.Nielsen. 2015).

Las redes LSTM son un caso particular de las recurrentes. Estas últimas se caracterizan por tener una estructura de retorno o bucle, es decir, interactúan con nodos anteriores. La función de estos es de memoria, tomando en cuenta así la información actual y la anterior. Las Redes Neuronales Recurrentes tienen memoria del pasado inmediato. (Niklas Donges, 2018).

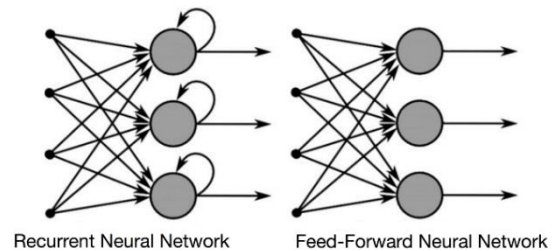


Gráfico N° 1. Diferencia de redes neuronales recurrentes y redes Feed Forward. Fuente: Niklas Donges. (2018)

Las LSTM, al ser una versión de las recurrentes, presentan un comportamiento similar, su diferencia principal es que conservan memoria importante a mayores plazos de tiempo. La caracterización se hace también a través de pesos que se adquieren mediante el entrenamiento de la red. También soluciona el problema de las Redes Recurrentes estándar de explosión de gradiente (los pesos incrementan de manera desmesurada) y desaparición de gradiente (los pesos desaparecen).

2.5. Ingeniería de características

Mejor conocido como "Feature Engineering" es el proceso mediante el cual, de los datos crudos se extraen características relevantes para la solución de un problema que puede ser procesado con métodos de machine learning. En estos métodos la extracción de características de los datos es de suma importancia, debido a que los algoritmos son muy sensibles a la información que se les provee por lo que, el uso de variables o información irrelevante puede distorsionar altamente el modelo.

2.6. Sistema de Variables

Para el índice del dólar, índice del euro y cambio del dólar/yuan se utilizaron la fecha (día, mes y año) y las variables clásicas OHLC (Open, High, Low, Close) que miden el valor de apertura, valor más alto, valor más bajo y cierre en un periodo de tiempo, que en nuestro caso es un día. Estas variables se utilizan porque captan la mayor parte de información del comportamiento de los individuos durante el periodo de tiempo. Del petróleo solo se tomó el precio diario debido a que era la única variable de la que se tenían datos confiables. A los individuos en estudio, los del mercado de criptomonedas, se les midió adicionalmente volumen diario y la capitalización de mercado, una variable importante ya que mide la liquidez del mercado, variables que no están presente en los índices y que no consideraremos en el cambio dólar/yuan ya que se asume una liquidez absoluta y no posee capitalización. Las variables respuesta serán los valores promedios de OHLC de las criptomonedas en estudio.

3. Metodología

Esta investigación es de nivel explicativo debido a que se busca probar si variables económicas tienen un efecto significativo en el desempeño de una red neuronal respecto a la predicción del precio de las

criptomonedas seleccionadas y se comparará el desempeño con el modelo ARIMA. Todos los datos de las variables son recolectados desde el 01/01/2015 hasta el 28/02/2019 tomados diariamente con un total de 1520 datos por variable tomando en cuenta que algunas criptomonedas iniciaron después de esa fecha por lo que se obtuvo menos información de las mismas. Estos fueron tomados de las siguientes páginas confiables de internet: CoinMarketCap, Investing, Stooq y Quandl.

3.1. Los datos

Los datos faltantes fueron de las variables no pertenecientes al mercado cripto debido a que no cotizan los fines de semana, por lo que se asumieron sus valores iguales al de los días de cierre. Se utilizó la herramienta programática Python y sus paquetes: Keras para las redes neuronales y Statsmodels para el modelo ARIMA además de librerías complementarias. Para usar la librería de Keras, los datos deben transformarse en un problema de aprendizaje supervisado. Básicamente, se reordenan las variables en 2 grupos: unas en el tiempo t y otras en el tiempo $t-j$. En este caso usaremos solo $t-1$. Por lo que una base de datos de p variables pasa a tener $2p$ variables donde para cada variable i se añade una j en el tiempo $t-1$.

El entrenamiento y las predicciones se harán sobre datos escalados entre 0 y 1, esto debido a que, como se explicó anteriormente, es una pequeña aplicación de ingeniería de características para acelerar el aprendizaje de las redes. Luego el conjunto de datos debe dividirse entonces en un conjunto de entrenamiento y otro de prueba. Los resultados luego serán re-escalados a sus valores iniciales para así hacerlos comparables, y medir la precisión de las predicciones con el estadístico y los índices.

3.2. Los índices

Se utilizó el índice MAPE (Mean Absolute Percentage Error) que mide error porcentual, el cual se calcula:

$$MAPE = \frac{100}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{|y_i|}$$

Donde y_i es el i -ésimo valor real y \hat{y}_i el i -ésimo valor estimado. Su fácil interpretación lo hace un índice frecuentemente utilizado. En la medida que disminuye su valor, el modelo presenta menos error.

Se usó también el índice MASE (Mean Absolute Scaled Error) el cual es conveniente ya que está diseñado para series de tiempo, además de tener propiedades deseables como ser invariante de escala, simétrico y de fácil interpretación, se ha demostrado empíricamente que, para este índice, el estadístico Diebold-Mariano (DM) se distribuye asintóticamente normal estándar para contrastar la diferencia significativa entre las predicciones de 2 modelos (Franses, 2016). Este índice se calcula de la forma:

$$MASE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{\frac{n}{n-1} \sum_{i=2}^n |y_i - y_{i-1}|}$$

Donde y_i es el i -ésimo valor real y \hat{y}_i el i -ésimo valor estimado. Su interpretación es: $MASE > 1$ implica que el pronóstico actual es peor que un método primitivo o Navie Forecast y si $MASE < 1$ implica que el pronóstico actual es mejor que un método primitivo,

3.3. El estadístico

El estadístico Diebold-Mariano se calcula de la siguiente manera:

$$DM = \frac{\overline{d_{12}}}{\hat{\sigma}_{d_{12}}} \sim N(0,1)$$

Donde: $\overline{d_{12}}$ es el promedio de la diferencia de las 2 funciones de pérdida y se define como $\frac{1}{n} \sum_{j=1}^n (l_{1j} - l_{2j})$ donde, en este caso l_{1j} y l_{2j} son los n valores de las n diferencias de la función de pérdida MASE

para cada modelo. $\hat{\sigma}_{d_{12}}$ Es la estimación de la desviación estándar del promedio de la diferencia de las funciones de pérdida. Proporcionando el valor crítico para contrastar la hipótesis de que la diferencia de las funciones de pérdida es 0, es decir: $H_0: d_{12} = 0$

Con ambos índices (MASE y MAPE) y el test de hipótesis se buscó obtener resultados consistentes que determinaron si había o no diferencia entre los modelos.

3.4. La red neuronal

Se debe entender que no existen formas específicas de determinar los mejores hiperparámetros de una red neuronal. Para obtener los mejores resultados posibles primero se ajustarán varios modelos plausibles. Los modelos considerados son los siguientes:

- modelo 1: 1 capa de entrada, 1 capa internas LSTM con 0.3 dropout, 1 capa LSTM sin dropout, 1 capa de salida
- modelo 2: 1 capa de entrada, 2 capas internas LSTM con 0.3 de dropout cada una, 1 capa LSTM sin dropout, 1 capa de salida
- modelo 3: 1 capa de entrada, 3 capas internas LSTM con 0.3 de dropout cada una, 1 capa LSTM sin dropout, 1 capa de salida
- modelo 4: 1 capa de entrada, 1 interna LSTM sin dropout, 1 capa de salida.

La capa de entrada tiene tantos nodos como variables (1 para univariante y 20 para multivariante) las capas internas tienen 50 nodos LSTM de las cuales, las que tienen función DropOut al 0.3 desactivan el 30% de las neuronas de manera aleatoria para crear nuevos caminos de entrenamiento para las neuronas y evitar el sobre-entrenamiento.

Por la razón mencionada anteriormente, a través de ensayo y error, se pudo observar que valores tentativos para los valores de epoch, el cual se refiere a el número de

iteraciones de aprendizaje de los datos en la red neuronal y batch size que se refiere al número de datos procesados antes de actualizar los valores de la red, son 150 y 600 respectivamente.

3.5. EL modelo ARIMA

Para los datos históricos del promedio de las variables Open-High-Low-Close (OHLC), se hizo un análisis de series de tiempo con ARIMA. Se estudió su estacionariedad de manera gráfica, así como sus auto correlaciones y auto correlaciones parciales para encontrar el orden más plausible de los componentes AR (Auto regresivo) y MA (Media móvil). Se estimó finalmente el modelo ARIMA que mejor resultó ajustarse a los datos de las criptomonedas. También se estudiaron las densidades residuales y correlaciones residuales para constatar que el modelo posee características deseables que lo hacen válido.

El cálculo del estadístico DM así como los índices se realizaron igualmente para estos modelos. Estos se hicieron con las herramientas de la librería statsmodels de python directamente con los datos originales.

4. Resultados y discusión

4.1. Modelos LSTM

Para cada criptomoneda se ajustarán 4 modelos posibles. Estos modelos se comparan entre sí para determinar cuál de ellos representa mejor la serie, el modelo que mejor se ajuste a los datos multivariantes será el contrastado contra el mejor modelo para la serie univariante, obteniendo así un resultado más orientado a la eficiencia de las variables que a la de los modelos. Habiéndose ajustado la mejor cantidad de epochs y el mejor batch size, se procedió a graficar y estimar los valores predichos sobre cada serie para cada modelo, para así compararlos con sus homólogos a través del estadístico DM y los índices MASE y MAPE.

Entre los modelos multivariantes (que consideran las variables económicas) y los univariantes, tuvieron mejor desempeño todos los modelos univariantes, esto determinado a través de las mediciones antes mencionadas (Cuadro N° 1).

Primeramente, puede observarse que muchas veces el modelo con más capas no fue siempre el más efectivo sino otros como el modelo 1 que contiene solo 2 capas LSTM (una con dropout) e incluso el 4 con solo una capa LSTM sin dropout ambos casos ejemplificados en el Bitcoin.

Como se ha estudiado antes, el número de capas aumenta la complejidad de los patrones reconocidos por una red neuronal, pero está sujeto también al tamaño y calidad del conjunto de entrenamiento debido a que más capas implica más parámetros, lo que requiere más información. (Larrañaga P. et al.,2019)

Cuadro N° 1. Comparación de los modelos multivariantes y univariantes por criptoactivo

Cripto activo	Variable	Modelo	MASE	MAPE	DM de la diferencia	p-valor de la diferencia
Bitcoin	M	4	0.59	1.13	1.72	0.08
	U	1	0.34	0.65		
Lite coin	M	3	1.71	4.90	4.07	0.00**
	U	2	0.30	1.05		
XRP	M	1	0.38	0.85	2.78	0.00**
	U	1	0.06	0.15		
Bitcoin Cash	M	4	3.52	54.18	3.62	0.00**
	U	3	1.77	42.01		
ETH	M	1	0.59	1.70	1.87	0.06
	U	3	0.51	138		

M = modelo multivariante. U = modelo univariante
 ** = diferencia significativa para todo nivel de confianza. Fuente: elaboración propia

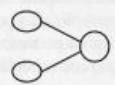
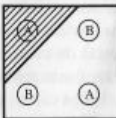


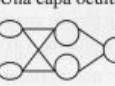
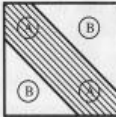



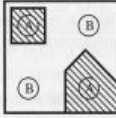

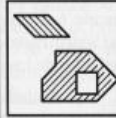
Arquitectura	Región de decisión	Ejemplo 1: XOR	Ejemplo 2: clasificación	Regiones más generales
Sin capa oculta 	Hiperplano (dos regiones)			
Una capa oculta 	Regiones polinomiales convexas			
Dos capas ocultas 	Regiones arbitrarias			

Gráfico N° 2. Regiones de decisión. Nota. Regiones de decisión obtenidas para el perceptrón simple (arriba), el perceptrón multicapa con una capa oculta (en medio) y el perceptrón multicapa con dos capas ocultas (abajo). Fuente: Larrañaga P. *et al.* (2019)

Es posible que las redes más complejas pudieran superar a las redes sencillas, pero en este caso limitado con una cantidad de datos fija una red más sencilla pero mejor entrenada tuvo superioridad en algunos casos. Particularmente se ve reflejado en la serie del BitcoinCash, que con la poca información disponible hasta la fecha arrojó resultados incluso peores que un modelo ARIMA, probablemente debido a la falta de datos.

Es muy interesante notar que para los modelos LSTM, los que consideraron solo los datos históricos obtuvieron en general mejores resultados, aun cuando en estadística tiende a mencionarse que agregar variables a un modelo (incluso variables espurias) incrementarán la variabilidad explicada así sea de manera ficticia.

La razón del bajo desempeño en las predicciones de los modelos multivariantes podría tener varias posibles respuestas: Una es debido a la forma de aprendizaje de las redes neuronales, que buscan captar el cambio de una variable con respecto a otra (Michael, A.Nielsen, 2015), podría ser que

simplemente las variables de los individuos propuestos no están suficientemente correlacionados con el mercado de criptoactivos como para mejorar las estimaciones de los modelos.

Cabe destacar el análisis exploratorio de los datos, en el cual se observó que alrededor de las fechas de la burbuja económica de las criptomonedas el petróleo y el índice del euro estaban en alza, y el índice del dólar y el cambio dólar yuan estaban en picada, sosteniendo valores particularmente altos y particularmente bajos de esas series, es posible que estas variaciones se debieran a mero ruido estocástico o debido a situaciones externas no relacionadas con el mercado de criptodivisas. Cabe acotar que las burbujas económicas son situaciones en las que los precios de un mercado suben debido a la especulación y no al valor real del activo (Gordon Scott, 2019), por lo que los precios reales de las criptomonedas no se ven reflejados para esas fechas, y al agregar falsas tendencias de las variables económicas se termina desorientando el modelo dando como resultado peores predicciones con más variables a la hora de probar el modelo entrenado.

Cabe notar que los modelos ajustados para la serie BitcoinCash, tuvieron predicciones muy pobres en general, con y sin las variables económicas a diferencia de las demás series. De esto se desprende que, aunque limitada, la selección de modelos se ajustó a los datos y que este caso particular como mencionamos anteriormente se debió a falta de información.

Siendo así, que los modelos univariantes en general captarán mejor la variabilidad, indica en primera instancia la posibilidad de que los individuos no están significativamente relacionados como para incrementar la variabilidad del modelo, o en última instancia y con seguridad, que las variables económicas escogidas no son significativas en el estudio.

Una segunda posibilidad es que incluir los datos de la burbuja económica reduce la captación de la variabilidad compartida

que estas variables económicas puedan tener con el mercado de divisas.

En pocas palabras, la complejización del modelo no obtuvo ganancias con respecto al valor informativo obtenido y aunque este puede deberse a varios factores ya mencionados, el más plausible es la no relevancia de las variables planteadas.

4.2. Modelos ARIMA

Utilizando los valores reales (no transformados) se ajustaron y estimaron los mejores modelos ARIMA para cada serie de tiempo de las criptomonedas (Cuadro N° 2).

A primera vista se observa que el modelo ajustado al BitcoinCash es el único que es superior al modelo univariante ajustado con redes neuronales LSTM.

Cuadro N° 2. Modelos ARIMA y sus rendimientos

Cripto Activo	Modelo	MASE	MAPE
Bitcoin	ARIMA (1,1,1)	0.89	1.43
Litecoin	ARIMA (2,1,1)	0.85	2.62
XRP	ARIMA (3,1,2)	1.33	2.88
BitcoinCash	ARIMA (1,1,1)	0.92	3.53
Ethereum	ARIMA (1,1,2)	0.96	2.67

Fuente: elaboración propia

4.3. LSTM vs ARIMA

El trabajo de redes neuronales requiere una capacidad enorme de cómputo a diferencia de los modelos ARIMA. El total de tiempo de procesamiento para la obtención de los datos para las redes LSTM de manera conjunta fue superior a las 2 horas, mientras que para el modelo ARIMA fue inferior a 5 minutos. Esto se debe a la complejidad de

las redes neuronales en contraste con la sencillez de los modelos ARIMA. El principal aprendizaje que deja esta comparación es: dejar soluciones complejas a problemas complejos.

Distintos autores como Karakoyun & Cibikdiken (2018) y Anshul & Sukumar (2018), han señalado en sus trabajos la superioridad en las predicciones de los modelos LSTM sobre los ARIMA. Aunque empírica, la superioridad parece ser consistente en la mayoría de los casos incluyendo la presente investigación. Es curioso observar el caso de la serie del BitcoinCash que se sale de esta norma. Esto es debido a que la sencillez de los modelos ARIMA le permite ajustarse con una cantidad mucho menor de datos. Esto hace notar que para conjuntos de datos pequeños las redes neuronales no son potentes y modelos tradicionales podrían ajustarse mejor

Existen métodos que orientan a la selección del mejor modelo ARIMA guiados por el comportamiento de los datos como Cryer j. & Chan K. (2008) mencionan en su investigación, mientras que la selección de las características de los modelos LSTM debe aprenderse con la práctica y la experiencia de personas más experimentadas tanto en el mundo de las redes como del problema en sí.

Sin embargo, vimos que para el problema planteado bastó con probar 4 modelos para obtener uno mejor que ARIMA por lo que el proceso de ensayo y error no es tan extenuante como podría pensarse. Aunque a los modelos ARIMA se le ajustan de manera previa los datos, no hay reglas exactas para este proceso tampoco. Como demuestra Rob Hyndman (2019), programador de paquetes de R, el cual actualmente es el programa estadístico con más funciones implementadas en la actualidad, en la implementación del paquete auto.arima que básicamente prueba modelos ARIMA en un rango determinado para p, d y q seleccionado para hallar el mejor modelo entre todos.

Sin duda alguna el experimento proveyó información sobre la no eficiencia de las

variables, y su valor informativo con respecto a su uso en el ámbito de las criptomonedas sigue siendo significativo.

5. Conclusiones

La composición óptima de las redes neuronales se logró probando entre 4 arquitecturas distintas obteniendo así mejores resultados y orientándolos hacia la eficiencia de las variables y no del modelo por lo que a la hora de componer estos algoritmos deben probarse varias arquitecturas para obtener resultados óptimos.

Las variables económicas tuvieron un efecto negativo con respecto a las predicciones de los cryptoactivos elegidos, solo con una diferencia no significativa para todo nivel de confianza. En todo caso, no mejoró las predicciones, por lo que dichos individuos o al menos las variables consideradas de estos para el presente estudio no aportaron información relevante. La utilización de estas variables no aporta ninguna mejora a los modelos considerados. Obteniendo también como resultado una superioridad de las redes LSTM univariantes debido a la distorsión que aportaron las variables económicas consideradas.

El modelo LSTM superó a los modelos ARIMA de manera consistente exceptuando un caso, el del BitcoinCash, que reveló la importancia de la cantidad de información a la hora de entrenar una red neuronal, por lo que, para ese caso particular, el modelo ARIMA sería más eficiente y rápido.

Para ciertos métodos de ML es menester la disposición de equipos adecuados y capaces debido a la alta demanda computacional de estos métodos.

6. Referencias

- Anshul S., T.R. Sukumar. (2018). *Predicting bitcoin price using LSTM and Compare its predictability with ARIMA model. International Journal of Pure and Applied Mathematics Vol. 119: (17) pp 2591-2600.* Recuperado de: <https://acadpubl.eu/hub/2018-119-17/3/214.pdf>
- Antoniadis, I., Sariannidis, N., and Kontsas, S. (2018). *The Effect of Bitcoin Prices on US Dollar Index Price. Springer Proceedings in Business and Economics*, pp 511–521.
- Azari, A. (2018). *Bitcoin Price Prediction: An ARIMA Approach. KTH Royal Institute of Technology.* Recuperado de: https://www.researchgate.net/publication/328288986_Bitcoin_Price_Prediction_An_ARIMA_Approach
- Coinmarketcap.(2019). *Historical Snapshot - February 24, 2019.* Recuperado de: <https://coinmarketcap.com/historical/20190224/>
- Coinmarketcap B. (2019). *Global Charts - Percentage of Total Market Capitalization (Dominance)* . Recuperado de: <https://coinmarketcap.com/charts/#dominance-percentage>
- Cryer J., Chang K. (2008). *Time Series Analysis With Applications in R. Springer. Second Edition.*
- Franses, P. H. (2016). *A note on the Mean Absolute Scaled Error. International Journal of Forecasting*, 32(1), 20–22. Recuperado de: <https://sci-hub.tw/10.1016/j.ijforecast.2015.03.008>
- Gordon Scott. (2019). *Bubble Theory. Investopedia.* Recuperado de: <https://www.investopedia.com/terms/b/bubble-theory.asp>
- ICE Futures U.S. (2015). *U.S. Dollar Index Contracts - FAQ.*
- Investing. (2019). *US Dollar Index (DXY).* Recuperado de: <https://www.investing.com/indices/usdollar>
- Investing B. (2019). *USD/CNY - US Dollar Chinese Yuan.* Recuperado de:

<https://www.investing.com/currencies/usd-cny>

Investing Support. (2019). *Where do you get your data quotes from?*. Investing. Recuperado de: <https://www.investing-support.com/hc/en-us/articles/115004038165-Where-do-you-get-your-data-quotes-from>

Karakoyun, E. and Cibikdiken, A. (2018). *Comparison of ARIMA Time Series Model and LSTM Deep Learning Algorithm for Bitcoin Price Forecasting*. Ch p. 171-179. En: Vopava, J., Douda, V., Kratochvil, R., and Konecki, M. Editores. *Proceedings of MAC 2018 in Prague*. Academic Conferences Association, z.s. Praga-República Checa.

Michael, A.Nielsen. (2015). *Neural Network and Deep Learning*. Determination Press. Recuperado de: <http://neuralnetworksanddeeplearning.com/index.html>

Niklas Donges. (2018). *Recurrent Neural Networks and LSTM*. Medium. Recuperado de: <https://towardsdatascience.com/recurrent-neural-networks-and-lstm-4b601dd822a5>

Larrañaga P., Inza I., Moujahid A. (2019). *Redes Neuronales*. Departamento de Ciencias de la Computación e Inteligencia Artificial. Universidad del País Vasco-Euskal Herriko Unibertsitatea. Recuperado de: <http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t8neuronales.pdf>

Organization of Petroleum Exporting Countries (OPEC). (2017). *OPEC Share of World Crude Oil Reserves, 2017*. Recuperado de: https://www.opec.org/opec_web/en/data_graphs/330.htm

Quandl. (2019). *OPEC Crude Oil Price*. Recuperado de: <https://www.quandl.com/data/OPEC/ORB-OPEC-Crude-Oil-Price>

Rob Hyndman. (2019). *auto.arima*. R Documentation. Recuperado de: <https://www.rdocumentation.org/packages/forecast/versions/8.7/topics/auto.arima>

Roy, S., Nanjiba, S., & Chakrabarty, A. (2018). *Bitcoin Price Forecasting Using Time Series Analysis*. 2018 21st International Conference of Computer and Information Technology (ICCIIT).

Stooq. (2019). *Historical data: Euro Index (EUR_I)*. Recuperado de: https://stooq.com/q/d/?s=eur_i