Time series forecasting based on neural networks and technical analysis indicators

Jeison Ivan Roa Mora, Diego Alejandro Barragán Vargas
Maharishi international University, Universidad Distrital Francisco Jose de Caldas
jeisonroa1@miu.edu,dabarraganv@correo.udistrital.edu.co

ABSTRACT

This research evaluates the performance of different prediction models implemented with artificial neural networks by means of a technique called LSTM and using ARIMA models. For this purpose, error indicators such as RMSE, MAPE, MAE, MPE and correlation coefficient are used to evaluate their performance. The main objective of these prediction systems is to know the future value of the exchange value of a currency pair. In this case we will work on BTC/USD and XRP/USD datasets. All the implementation will be done in Python using libraries such as Pandas, numpy, scikit learn, keras and TensorFlow.

KEYWORDS

Arima, artificial neural networks, crypto, forecasting, time series, economics.

Reference Format:
Jeison Ivan Roa Mora, Diego Alejandro Barragán Vargas. 2023. Time series forecasting based on neural networks and technical analysis indicators. In *Proceedings of* . 11 pages. https://doi.org/XX.XX/nnnnnnn.nnnnnnn

1 INTRODUCTION

With the rise of neural networks applied to the resolution of prediction problems, it is considered important to continue researching in this promising line of solving problems that previously seemed unsolvable for machines. That is why we propose to implement and test different neural models in Python language and compare their performance with statistics models such as ARIMA, which has been shown to perform well in stationary time series. That is why the first thing that will be sought in these series is to make them stationary using transformations of various types. One of the novelties in this article is the use of economic indicators as inputs for LSTM networks as well as income as input of past values, making an additional recurrence to the one that LSTM inherently includes by definition.

2 THEORETICAL BACKGROUND

2.1 Bitcoin and Ripple

Bitcoin is one of the cryptocurrencies with the largest market capitalization and XRP is an emerging cryptocurrency that has had good volatility in its value. Both share the principle of being a decentralized currency, so there are no authorities or control entities that regulate them in terms of issuing money or recording their transactions, in addition to generating excellent returns in a short time when a proper analysis is made. A central point of analysis is the behavior of its value over time through different series and parameters that provide correct information about the future values it could take. (1)(2) (3) (4).

Bitcoin was launched in 2008 and is currently the most valuable currency within cryptocurrencies, this is largely because it was designed as a digital currency system based on peer-to-peer networks which gave it greater autonomy and participants using the currency can have a distributed network, adding its high reliability as all transactions are verified by complete nodes and are stored in blocks that are chained by associating a hash of previous block header. Each block contains a merkle root that in simple words can be considered as a digital fingerprint, which prevents criminals from manipulating the block chain at will, allowing the btc to have in short properties of immutability, decentralization, data integrity and robust security, which allows it to be considered a reliable digital currency. (5).

2023. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XX.XX/nnnnnnn.nnnnnnn

The XRP currency is a free software project and payment protocol that pursues the development of a credit system based on the peer-to-peer paradigm. Each node in the network functions as a local exchange system, so that the whole system forms a decentralized mutual bank. Some consider that taken to its ultimate consequences, this network would constitute a decentralized social network service based on honor and trust among its participants worldwide (thus, its financial capital is supported by social capital). A scaled-down version of the network would consist of an extension of the traditional banking system in which there would be alternative payment channels that would not depend on central banks. The launch year of the currency was 2012. (3) (6).

2.2 Basic Concepts of ARIMA Models

2.2.1 Stochastic Process: A stochastic process is a succession of ordered random variables Yt, where t can take any value between $-\infty and \infty$. It is important to note that the subscript t in a stochastic process referred to time series, represents the time step, as well as each of the variables Yt, which compose the stochastic process, will have its own distribution function with its corresponding moments, as well as each pair of variables will have its corresponding joint distribution function and its marginal distribution functions. (7).

2.2.2 Stationary Stochastic Process: A stochastic process is stationary if the joint distribution functions are invariant with respect to a shift in time (variation of t). Where t, t+1, t+2, ..., t+k are considered to reflect successive periods as shown in the following equation:

$$F(Y_t, Y_{t+1}, ..., Y_{t+k}) = F(Y_{t+m}, Y_{t+m+1}, ..., Y_{t+k+m}) \quad (1)$$

where t,k and m can be any value.

2.2.3 Autoregressive models. A model is autoregressive if the endogenous variable of a period t is explained by its own observations corresponding to previous periods, adding an error term, as in structural models. In the case of stationary processes with normal distribution, the statistical theory of stochastic processes says that, under certain preconditions, all Yt can be expressed as a linear combination of its past values (systematic part) plus an error term (innovation).

2.2.4 Moving Average Model. A moving average model is one that explains the value of a given variable in period t from an independent term and a succession of errors corresponding to previous periods, suitably weighted. These models are usually denoted by the acronym MA, followed, as in the case of autoregressive models, by the order in parentheses.

2.3 ARIMA Model

In its most general form, the ARIMA(p,d,q), ARIMA(P,D,Q,)S model could be written as:

$$Y_T = \Phi_1 Y_{T-1} + \Phi_2 Y_{T-2} + ... + \Phi_{Ps+p+Ds+d} Y_{T-Ps-p-s-D-d} + \delta + U_t + \Theta_1 U_{T-1} + ... + \Theta_{Qs+q} U_{T-sQ-q} \quad (2)$$

Understanding that there may be more than one process generating the series (in the regular and seasonal parts) and writing a combination of the MA(q) and AR(p) models that have required a series of differentiations "d" in the regular part or "D" in the seasonal part to make them stationary (8).

2.4 Artificial Neural Networks

In 1943 McCulloch and Pitts designed the model of an artificial neuron, describing a computational mathematical model that attempted to simulate the behavior of a real biological neuron, this neural design is the basic processing element in an artificial neural network. (9) (10).

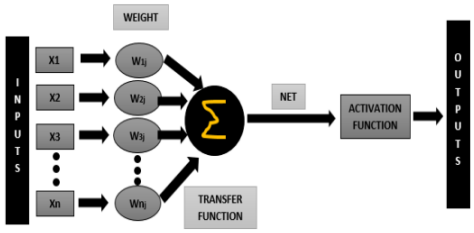

Figure 1: Structure of an artificial neuron (11).

The model is composed of a vector of weights W=(W1j, W2j, W3j...,Wnj), where W1j is the threshold of action or activation, in figure 1 we can observe the output of the neuron that for this case will be called Y, the activation function U, the inputs X=(X1,X2,X3,...,Xn), the mathematical model for a single neuron is shown below:

$$Y = U(\sum_{m}^{i=1} X_i W_i - W_0) \quad (3)$$

2.5 Long Short-Term Memory-LSTM

It is an extension of artificial neural networks, which allows expanding the memory of the network with the aim of learning important experiences that have taken place over a long time. LSTMs allow artificial neural networks to remember inputs over a long period of time (12) (13). This problem, called "long-term dependencies", was addressed by Hochreiter and Schmidhuber in 1997, where they proposed the so-called LSTM cell, which allowed improving the ability to remember

a standard recurrent cell by introducing the so-called "gate" in the cell (14) (15).Below is the original architecture of the LSTM:

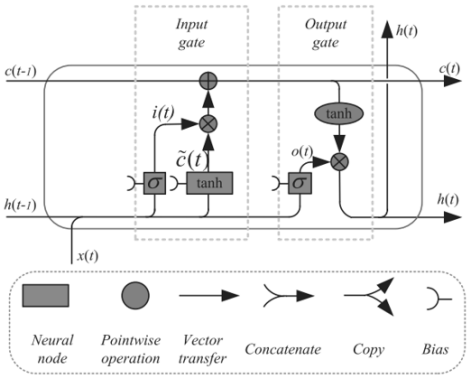

Figure 2: LSTM architecture. (14).

3 METHODOLOGY

Dataset

It is desired to use the BTC/USD and XRP/USD time series as the main information dataset for all the experimentation. Different pre-processing, training and validation processes will be performed for each of the cryptocurrencies to capture the different patterns and behaviors that each cryptocurrency naturally possesses.

The initial datasets were taken from cryptodatadownload.com using the Bitstamp cryptocurrency exchange data. The initial length of samples is 39547 for each dataset in which information can be found from 2018-05-15 to 2022-11-17.

The structure of the initial dataset is as follows:

(1) Date
(2) Unix
(3) Symbol
(4) Open
(5) High
(6) Low
(7) Close
(8) Volume BTC
(9) Volume USD

It is decided to use the post-pandemic information for the training of the systems because visually it has a bounded behavior that can be translated into a much higher prediction possibility based on some articles and information obtained during the previous research stage. The final dataset used in this paper is from 2018-05-15 to 2020-10-29 with a total of 18,000 samples. That is 18,000 BTC data and another 18,000 corresponding to XRP.

Data Preparation ARIMA

In the data preparation phase, first of all an exploratory analysis is performed where the distribution of the data can be observed as well as the initial graphs that were used to choose the data to be used for the training based on the graphs obtained and delimiting the post-pandemic data because the economy had a significant change and this was a fact that transformed the way the markets had behaved up to that moment. The following diagram shows the steps that will be carried out to develop these models:

Figure 3: ARIMA processing flow.

The following graphs show the behavior of the prices for both currencies in the original dataset.

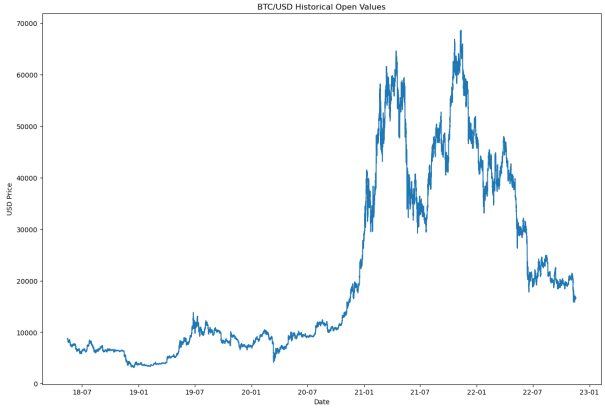

Figure 4: Original open values BTC/USD.

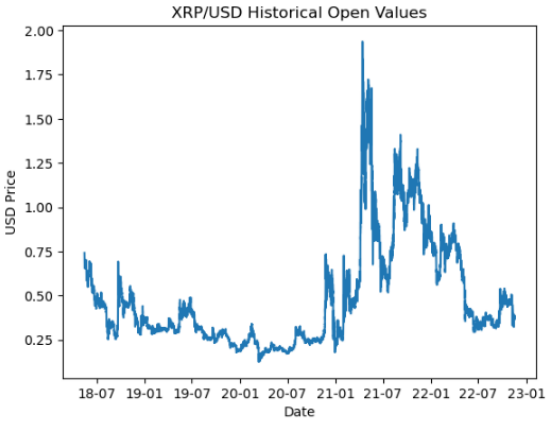

Figure 5: Original open values XRP/USD.

The following graphs show the dataset selected for training for both cases (18,000 records).

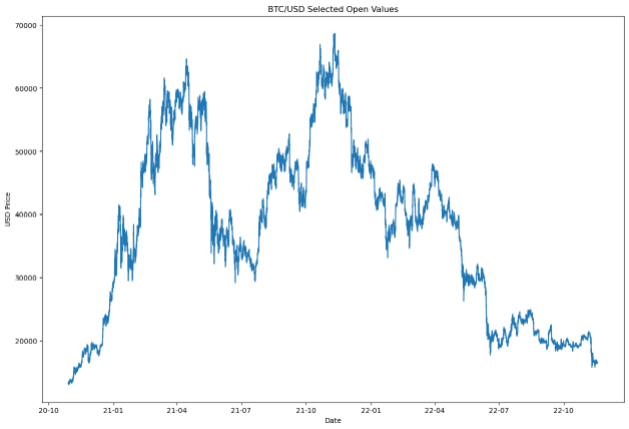

Figure 6: Selected open values BTC/USD.

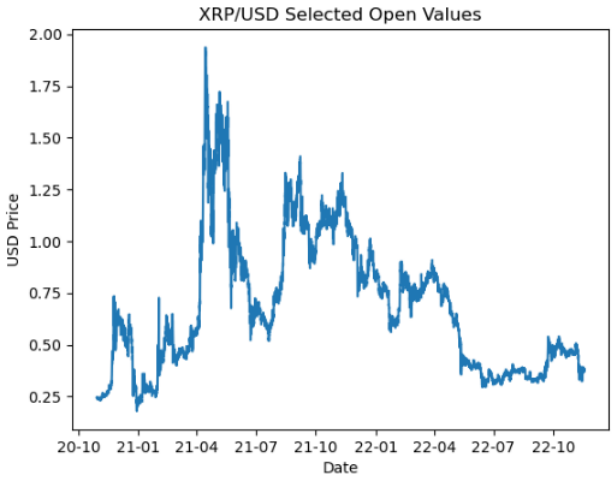

Figure 7: Selected open values XRP/USD.

It is worth mentioning that for the ARIMA case it was decided to use only the information from the open price column because the objective was not to use a multi-dimensional ARIMA model since ARIMA does not work very well in non-linear systems such as cryptocurrencies and doing a multi-dimensional analysis may not add much value, however it is a matter for future experimentation.

A sample of one of the final vectors used in the ARIMA fit is shown below:

```
--------------- Selected Open values (Pandemic times) ---------------
                        open
date
2022-11-18 00:00:00  16687.00
2022-11-17 23:00:00  16704.00
2022-11-17 22:00:00  16683.00
2022-11-17 21:00:00  16680.00
2022-11-17 20:00:00  16603.00
...                       ...
2020-10-29 05:00:00  13252.08
2020-10-29 04:00:00  13276.91
2020-10-29 03:00:00  13250.42
2020-10-29 02:00:00  13287.32
2020-10-29 01:00:00  13261.17

[18000 rows x 1 columns]
Total samples BTC/USD: 18000
```

Figure 8: ARIMA selected dataset BTC.

```
--------------- Selected Open values (Pandemic times) ---------------
                        open
date
2022-11-18 00:00:00  0.38148
2022-11-17 23:00:00  0.38360
2022-11-17 22:00:00  0.38314
2022-11-17 21:00:00  0.38229
2022-11-17 20:00:00  0.38038
...                      ...
2020-10-29 05:00:00  0.24587
2020-10-29 04:00:00  0.24691
2020-10-29 03:00:00  0.24547
2020-10-29 02:00:00  0.24608
2020-10-29 01:00:00  0.24515

[18000 rows x 1 columns]
Total samples XRP/USD: 18000
```

Figure 9: ARIMA selected dataset XRP.

One of the drawbacks during the development of this research was the fact that cryptocurrencies are highly non-linear systems and ARIMA works quite well with linear systems. Therefore, it was necessary to make an analysis of seasonality and filtering to identify trends and try to bring the dataset to a form as close to linear as possible. Below are the plots of the stationarity analysis as well as a resampling of the data set to reduce the high frequency noise and to visually highlight the low frequency component:

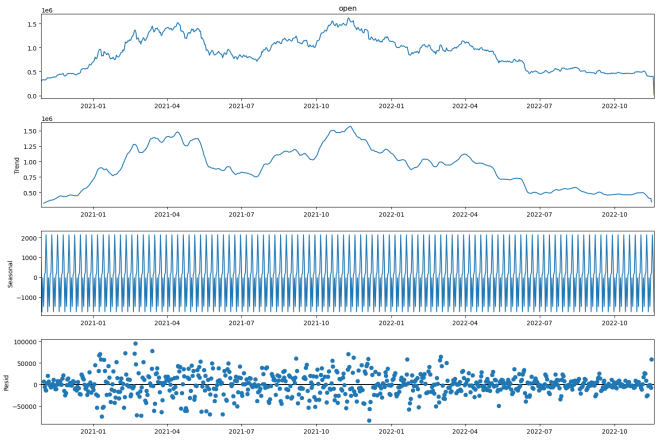

Figure 10: Seasonal descomposition BTC before transformation.

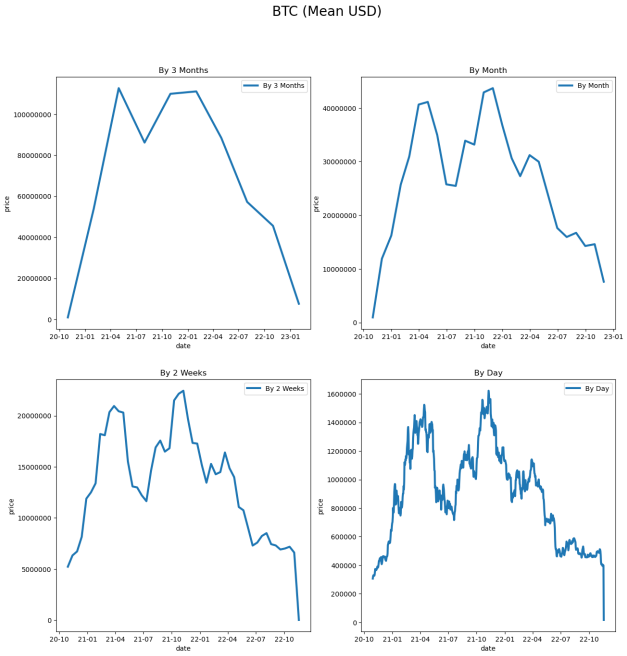

Figure 11: MEAN BTC/USD.

According to the charts below, the highest correlation occurs with a lag of 7, meaning that the data repeats a pattern on a weekly basis. This is somewhat true for the Bitcoin price, as from experience it has been seen that the price tends to have a similar behavior every week due to the periodicity of the markets and the actions of those involved.

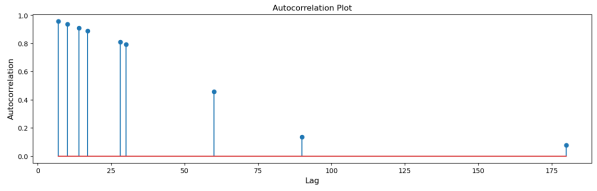

Figure 12: Autocorrelation Plot.

Another way of checking the autocorrelation is through Pandas tools library. Running that we can see positive correlation for the first 100 lags with the most significant ones being the first 10 ones.

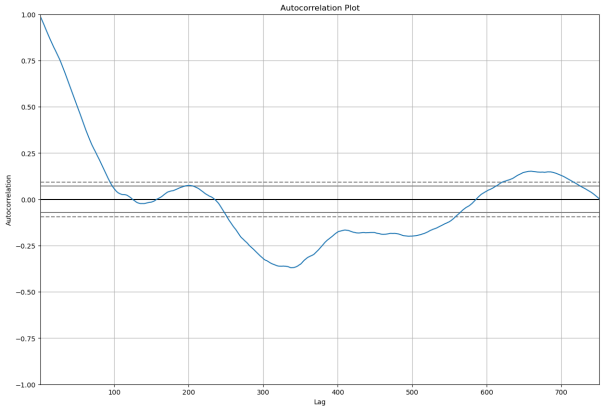

Figure 13: Autocorrelation Plot.

The Dickey-Fuller test was mainly used to find the degree of stationarity of the series. It became evident several times that the serie was not. To help the stationarity we first applied the Box Cox transformation, however, stationarity was not achieved in this way, so we progressively applied a Seasonal Differentiation and this in turn generated a set of data that complied with the Dicket-Fuller test and we proceeded to perform the training after this. Below you can see the plots of the seasonal decomposition as well as the final data after the transformation:

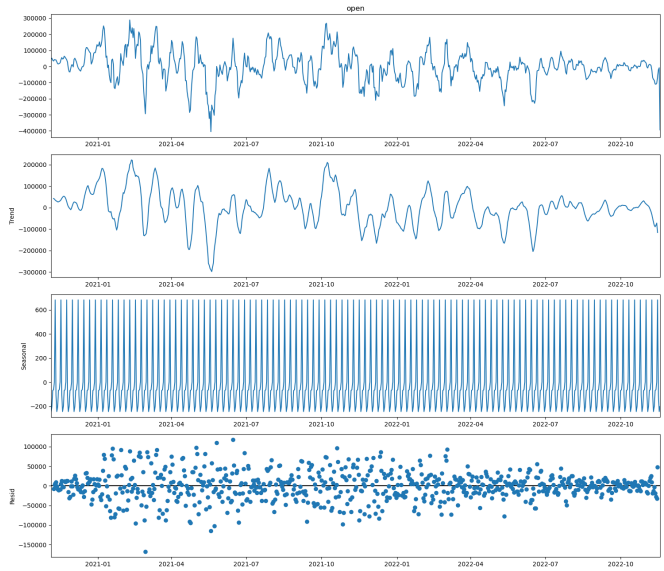

Figure 14: Seasonal descomposition BTC after transformations.

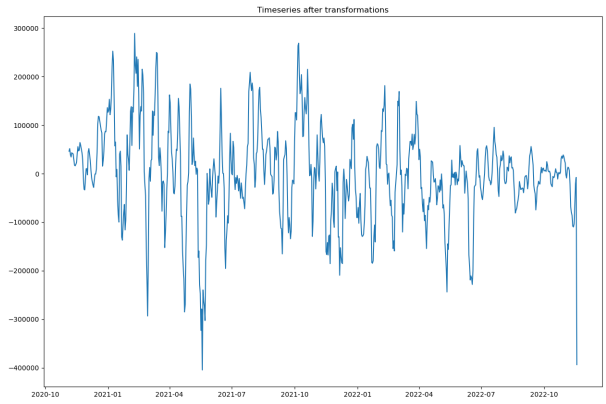

Figure 15: Time series after Transformations.

In the following flow chart it can be seen how this process was carried out where in the "Find p,d,q" box two methods were used, one automated and the other based on analysis. This is discussed in more detail in the next section.

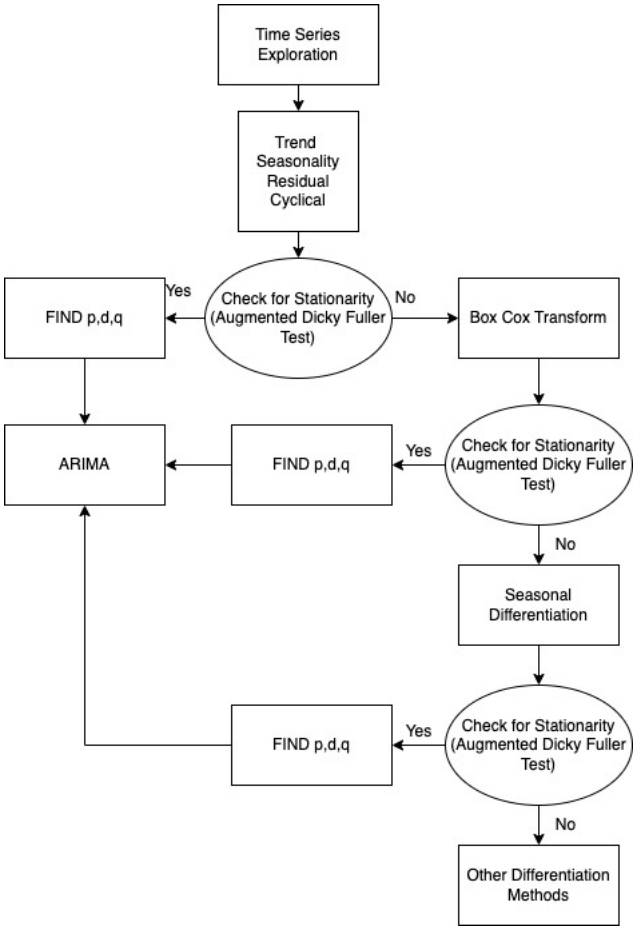

Figure 16: ARIMA Tunning

Data Preparation LSTM

In the case of the LSTM network, it is proposed to use as inputs several price steps including various technical indicators. The novelty in comparison to other studies found in the literature is that it has been proposed to add as input the last 4 values prior to the observation period, i.e. 5 consecutive price data are being entered for each set of inputs at a time K. A list of the inputs generated and used in this work is given below:

(1) Open price (k)
(2) Open price (k-1)
(3) Open price (k-2)
(4) Open price (k-3)
(5) Open price (k-4)
(6) BTC volumen
(7) RSI
(8) ROC

The following diagram shows the steps that will be carried out to develop these models:

Figure 17: LSTM processing flow.

After the organization of the initial matrix, a MinMax normalization is performed so that all values are bounded in the interval from 0 to 1. In this way the solution space is also bounded as it has been shown that better results can be obtained when dealing with neural networks.

The final structure of the processed dataset ready to be used for training and validation is shown below:

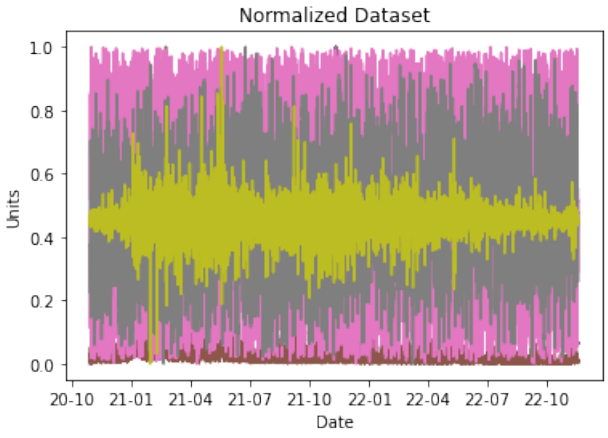

Figure 18: Normalized input dataset BTC

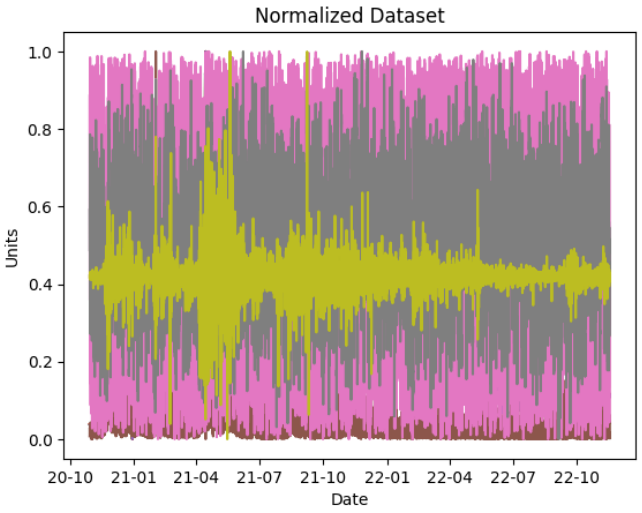

Figure 19: Normalized input dataset XRP

4 TRAINING AND RESULTS

4.1 ARIMA Models

For the ARIMA models the next step is to prepare the training and validation sets dividiendo los 18.000 datos y tomando el 90% para entrenamiento y dejando el 10% (180 Samples) para validacion y evaluacion de desempeno.

The first approach taken when fitting the ARIMA models was to rely on a library called auto_ARIMA, which allows to find the variables p,q,d that describe an ARIMA model in such a way that it is done automatically and not taking into account the correlation analysis mentioned above, with which it was found that 7 could be a good number for the lag.

The auto-ARIMA process seeks to identify the most optimal parameters for an ARIMA model, settling on a single fitted ARIMA model. This process is based on the commonly-used R function, forecast::auto.arima.

Auto-ARIMA works by conducting differencing tests (i.e., Kwiatkowski–Phillips–Schmidt–Shin, Augmented Dickey-Fuller or Phillips–Perron) to determine the order of differencing, d, and then fitting models within ranges of defined $start_p$, max_p, $start_q$, max_q ranges. If the seasonal optional is enabled, auto-ARIMA also seeks to identify the optimal P and Q hyper- parameters after conducting the Canova-Hansen to determine the optimal order of seasonal differencing, D.

In order to find the best model, auto-ARIMA optimizes for a given information criterion, one of ('aic', 'aicc', 'bic', 'hqic', 'oob') (Akaike Information Criterion, Corrected Akaike Information Criterion, Bayesian Information Criterion, Hannan-Quinn Information Criterion, or "out of bag"–for validation scoring–respectively) and returns the ARIMA which minimizes the value.

However, it was evidenced that the performance of the system proposed by auto-ARIMA turned out to be deficient in comparison to other sets of proposed parameters which presented better results. The following table shows the different experiments that were run to find the best hyperparameters. Error indicators such as MAE, MAPE, MPE, RMSE, and correlation coefficient were used.

The optimal model according to auto-ARIMA is model (1,1,2).

ARIMA P			ERROR INDICATORS				
p	d	q	MAE	MAPE	MPE	RMSE	Corr
0	1	0	16805.45	3.50	1.87	47388.66	0.55
1	1	1	16057.60	3.52	1.87	47250.25	0.56
1	1	2	16034.92	3.64	1.79	47100.69	0.56
5	1	0	16670.42	3.64	1.88	47042.57	0.56
5	2	0	18033.01	3.03	1.30	48831.11	0.54
5	2	1	16688.28	3.64	1.88	47058.39	0.56
7	1	7	13214.54	1.12	0.07	45046.34	0.61
7	1	14	13179.59	1.31	0.40	45089.64	0.61
7	2	7	13976.18	1.13	0.28	45081.84	0.61
7	2	14	13418.63	1.69	0.89	44898.02	0.61
8	2	8	13409.52	1.08	0.05	44155.65	0.63
8	3	8	13837.40	1.16	0.14	44520.08	0.62
9	1	9	13152.91	0.99	0.04	44856.53	0.61
9	2	9	13358.17	1.07	0.14	44776.71	0.61

Table 1: ARIMA BTC Experimental Results

The cells in yellow are the best performances for each of the performance indicators. Taking into account the RMSE and the correlation coefficient we can define that the best model is (8,2,8) which is not far from what was found in the processing stage because we did two transformations before meeting the Dickie Fuller criterion and the number for the lag found was 7, however with 8 a better result was achieved as shown above.

Some graphs of the results obtained with these models and their predictions are shown below:

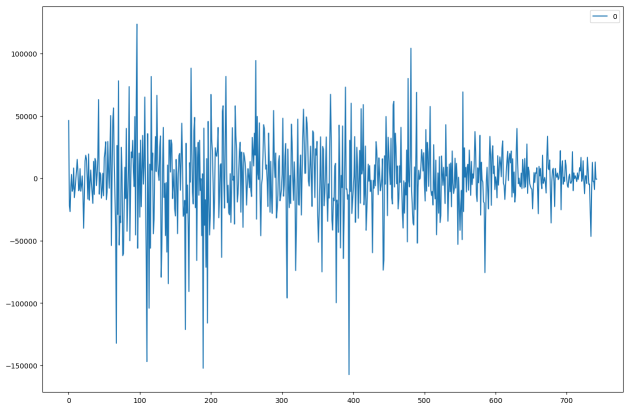

Figure 20: Residuals Plot BTC

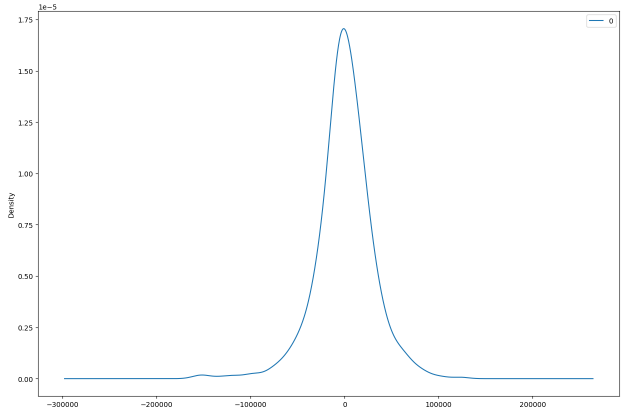

Figure 21: Residuals Density BTC

ARIMA P			ERROR INDICATORS				
p	d	q	MAE	MAPE	MPE	RMSE	Corr
0	1	0	0.56	1.68	0.31	1.20	0.68
1	1	1	0.55	1.71	0.15	1.19	0.69
1	1	2	0.55	1.70	0.16	1.19	0.68
2	0	4	0.50	1.36	0.03	1.19	0.68
5	1	0	0.55	1.72	0.16	1.20	0.69
5	2	0	0.61	1.96	0.01	1.26	0.67
5	2	1	0.55	1.72	0.15	1.20	0.68
7	1	7	0.42	1.24	0.11	1.08	0.74
7	1	14	0.44	1.23	0.01	1.10	0.73
7	2	7	0.45	1.33	0.15	1.09	0.74
7	2	14	0.43	1.28	0.14	1.08	0.74
8	2	8	0.45	1.33	0.10	1.08	0.75
8	3	8	0.47	1.61	0.08	1.11	0.74
9	1	9	0.43	1.29	0.11	1.08	0.74
9	2	9	0.43	1.27	0.10	1.08	0.75

Table 2: ARIMA XRP Experimental Results

For the case of XRP the result obtained by auto-ARIMA was the (2,0,4) model, however this model is far from being the one with the best performance. When analyzing the results it is obtained that the best model based on experimentation is the (9,2,9) model. This result is still consistent with previous analyses.

Some graphs of the results obtained with these models and their predictions are shown below:

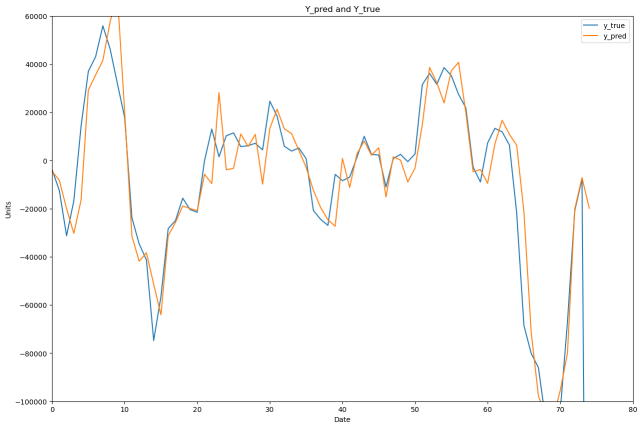

Figure 22: ARIMA predictions BTC

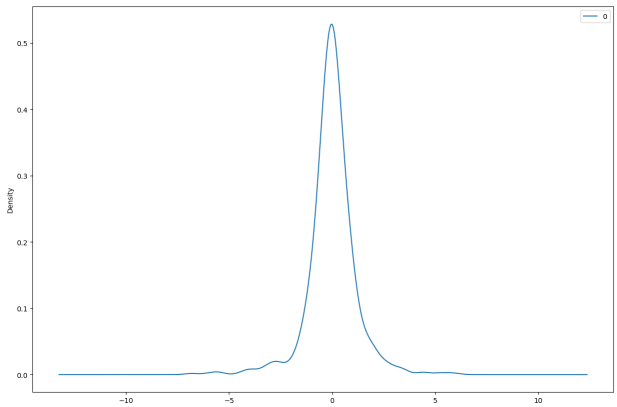

Figure 23: Residuals Plot XRP

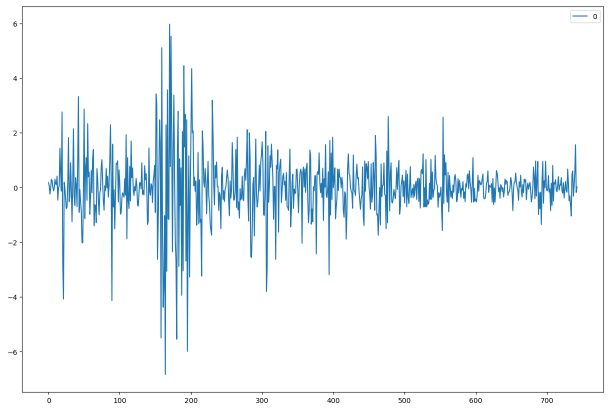

Figure 24: Residuals Density XRP

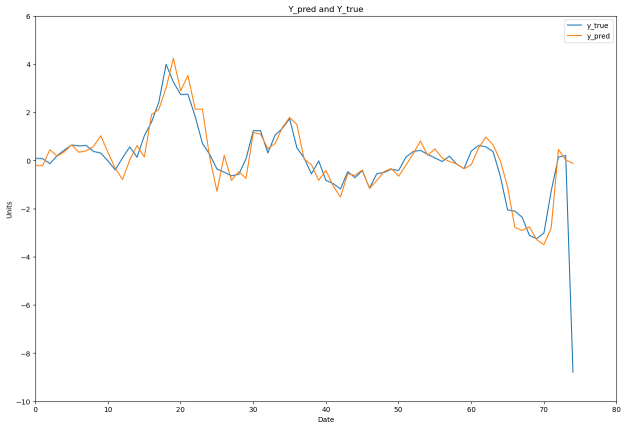

Figure 25: ARIMA predictions XRP

4.2 LSTM Models

For the case of LSTM networks, the following architecture has been used and is shown below:

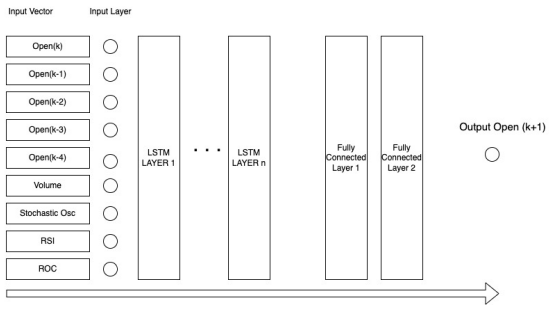

Figure 26: LSTM Architecture

Where the number of neurons in each LSTM layer is represented by the column N. L. and the number of neurons in the last fully connected layers is represented by the parameter F.C. It was decided to experiment by varying these values and thus find the optimal model with the proposed architecture. The experimental results are presented below.

HyperParam		ERROR INDICATORS				
N. L.	F. C.	MAE	MAPE	MPE	RMSE	Corr
2	128	490.143	0.029	0.029	514.144	0.655
2	64	21157.806	1.271	1.271	21158.262	-0.488
4	128	167.504	0.010	0.003	207.772	0.712
4	64	324.662	0.019	0.015	404.376	-0.134
8	128	461.347	0.027	0.027	488.317	-0.065
8	64	226.272	0.013	0.008	273.279	0.107
6	128	147.314	0.008	0.002	173.429	0.402
6	64	274.962	0.016	0.015	317.797	0.173
10	128	142.494	0.008	0.003	184.583	0.671
10	64	136.992	0.008	0.006	170.766	0.566

Table 3: LSTM BTC Experimental Results

As can be seen, a higher correlation was achieved with the model (4.128), however, better error minimization results were achieved for the model (10.64). What can be concluded from this is that probably the model (4.128) can be used for trend prediction or direction of movement rather than to predict the exact value of the price in the next period. If what is desired is information about the possible price that the currency will obtain in the next period it would be recommended to use the (10.64) model because although it has a lower correlation it has much smaller error indicators.

Some graphs of the results obtained with these models and their predictions are shown below:

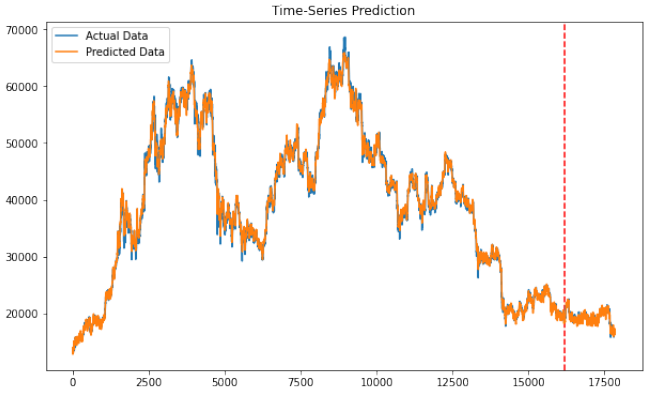

Figure 27: BTC Time series prediction

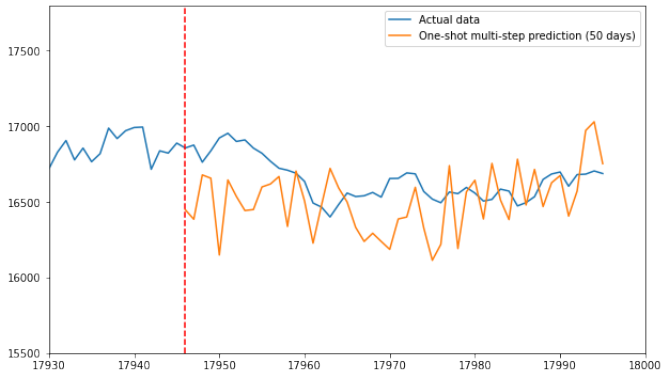

Figure 28: BTC Time series prediction zoom

HyperP		ERROR INDICATORS				
N. L.	F. C.	MAE	MAPE	MPE	RMSE	Corr
2	128	0.008	0.072	0.072	0.008	0.125
2	64	0.005	0.047	0.047	0.006	-0.060
4	128	0.007	0.019	0.001	0.009	0.646
4	64	0.007	0.020	0.008	0.008	0.590
8	128	0.011	0.029	0.029	0.012	0.449
8	64	0.014	0.039	0.039	0.016	0.569
6	128	0.006	0.017	0.015	0.008	0.622
6	64	0.005	0.013	0.004	0.006	0.692
10	128	0.005	0.013	-0.004	0.006	0.662
10	64	0.008	0.021	-0.019	0.009	0.690

Table 4: LSTM XRP Experimental Results

In the case of XRP, the model that provides the best correlation and therefore can be used for trend prediction is (6.64); however, the one that had the best performance in the other indicators was (10.128). Some graphs of the results obtained with these models and their predictions are shown below:

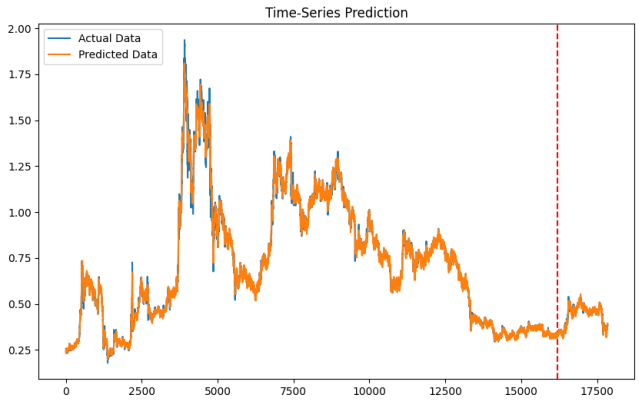

Figure 29: XRP Time series prediction

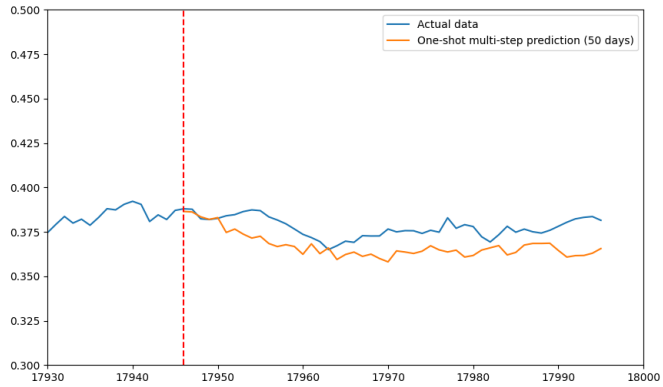

Figure 30: XRP Time series prediction

5 CONCLUSION

One of the most important conclusions found during the development of this article is that the ARIMA model does not present a good performance when it comes to time series prediction of cryptocurrencies because it is a highly nonlinear system and therefore it is necessary to apply certain transformations such as the Box Cox and the seasonal differentiation to make it linear, however the results are not comparable with those obtained for example with LSTM networks.

When the different normalizations such as Standard or MinMax were performed for the case of LSTM networks, a better performance was found using the MinMax transformation, which is consistent with what is found in the references studied for the development of this research.

It was found that the auto_arima library, although it avoids a theoretical analysis of the correspondence or correlation of the values, did not give very good results when evaluating the models proposed by this library. It is concluded that the traditional analysis for this specific problem may result in a better prediction performance, so for future work it is advisable not to use this optimization library.

By testing with different number of neurons and architectures in the LSTM network we found some optimal values where some models are more efficient for trend prediction while others seem to provide more value when trying to predict the discrete value at which the currency will open in the next period.

For future work we propose the use of Deeplearning using techniques such as convolutional neural networks that, although they have had a great boom in the use in image processing, could shed some light on advances in their use as discrete forecasting.

REFERENCES

[1] M. A. Ramírez, "Eficiencia en el mercado de criptomonedas," $https://fce2.ufm.edu/laissezfairenotes/static/media/Eficiencia_en_el_mercado_de_criptomonedas_Rev_MP_diagr.4dc6afc7019a10041fb5.pdf$, 2021, accedido 11-01-2022.

[2] J. A. C. Azofeifa, "Criptomonedas, ¿herramientas para el comercio de las multinacionales?" 2019, accedido 11-01-2022.

[3] J. M. CARRASCOSA, "Valor y futuro de las criptomonedas: Análisis crítico," $https://uvadoc.uva.es/bitstream/handle/10324/42041/TFG-J-181.pdf?sequence=1&isAllowed=y$, 2020, accedido 27-01-2023.

[4] C. Tandon, S. Revankar, H. Palivela, and S. S. Parihar, "How can we predict the impact of the social media messages on the value of cryptocurrency? insights from big data analytics," *International Journal of Information Management Data Insights*, vol. 1, no. 2, p. 100035, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667096821000288

[5] Y.-J. Lin, P.-W. Wu, C.-H. Hsu, I.-P. Tu, and S.-w. Liao, "An evaluation of bitcoin address classification based on transaction history summarization," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2019, pp. 302–310.

[6] S. Sridhar and S. Sanagavarapu, "Multi-head self-attention transformer for dogecoin price prediction," in *2021 14th International Conference on Human System Interaction (HSI)*, 2021, pp. 1–6.

[7] R. de Arce and R. Mahía, "Modelos arima," 2017, accedido 17-01-2023.

[8] ——, "Modelos arima," $https://gc.scalahed.com/recursos/files/r161r/w24503w/Box-Jenkins.PDF$, 2001, accedido 27-01-2023.

[9] R. A. Gonzalez, R. E. Ferro, and D. Liberona, "Government and governance in intelligent cities, smart transportation study case in bogotá colombia," *Ain Shams Engineering Journal*, vol. 11, no. 1, pp. 25–34, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090447919300851

[10] D. A. Barragán Vargas, R. A. Gonzalez Bustamante, and R. F. Escobar, "Artificial intelligence, case study: Detection of diabetic retinopathy through a neuronal networks in citizens of bogotá-colombia," in *Knowledge Management in Organisations*, L. Uden, I.-H. Ting, and B. Feldmann, Eds. Cham: Springer International Publishing, 2022, pp. 372–392.

[11] D. A. B. Vargas, R. F. Escobar, and J. E. S. Céspedes, "Classification of healthy persons and persons with ulnar syndrome by means of electromyographic signals feeding an artificial neural network," $https://www.researchgate.net/profile/Diego-Vargas-17/publication/354801400_Classification_of_Healthy_Persons_and_Persons_with_Ulnar_Syndrome_by_Means_of_Electromyographic_Signals_Feeding_an_Artificial_Neural_Network/links/614d53f0a595d06017e89587/Classification-of-Healthy-Persons-and-Persons-with-Ulnar-Syndrome-by-Means-of-Electromyographic-Signals-Feeding-an-Artificial-Neural-Network.pdf$, 2021, accedido 28-01-2023.

[12] K. Smagulova and A. P. James, "A survey on lstm memristive neural network architectures and applications." $https://doi.org/10.1140/epjst/e2019-900046-x$, 2019, accedido 28-01-2023.

[13] W. Zha, Y. Liu, Y. Wan, R. Luo, D. Li, S. Yang, and Y. Xu, "Forecasting monthly gas field production based on the cnn-lstm model," *Energy*, vol. 260, p. 124889, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544222017923

[14] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, 07 2019. [Online]. Available: https://doi.org/10.1162/neco_a_01199

[15] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid cnn-lstm model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180 544–180 557, 2020.