Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

NON-LINEAR AUTOREGRESSIVE WITH EXOGENEOUS INPUT (NARX) BITCOIN PRICE PREDICTION MODEL USING PSO-OPTIMIZED PARAMETERS AND MOVING AVERAGE TECHNICAL INDICATORS

N. I. Indera[1], I. M. Yassin[1,*], A. Zabidi[1] and Z. I. Rizman[2]

[1]Faculty of Electrical Engineering, UniversitiTeknologi MARA, 40450 Shah Alam, Selangor, Malaysia

[2]Faculty of Electrical Engineering, UniversitiTeknologi MARA, 23000 Dungun, Terengganu, Malaysia

Published online: 10 September 2017

ABSTRACT

This paper presents a Multi-Layer Perceptron (MLP)-based Non-Linear Autoregressive with Exogeneous Inputs (NARX) Bitcoin price forecasting model using the opening, closing, minimum and maximum past prices together with Moving Average (MA) technical indicators. As there were many parameter combinations to be tested, a Particle Swarm Optimization (PSO)-based method was used to optimize the number of hidden units, input lag and output lag of the NARX model. The results demonstrated the ability of the model to predict Bitcoin prices accurately while passing all model validation tests.

Keywords: Bitcoin price prediction; PSO; MLP; NARX; moving average; OSA.

Author Correspondence, e-mail: ihsan.yassin@gmail.com

doi: http://dx.doi.org/10.4314/jfas.v9i3s.61

1. INTRODUCTION

The stock market is a collection of markets and exchanges where the issuing and trading of stocks, commodities, derivatives and other sorts of securities take place. The stock market is a complex nonlinear dynamic system [1]. Notable amounts of research have been devoted to developing an efficient predictive model to assist traders in making wise investment decisions [2-3].

Market prediction is particularly difficult due to its complex nature [4-5]. However, the dynamics are not entirely unpredictable. Prediction involves analysis to predict the market direction using various methods such as statistical analysis, technical analysis, to fundamental analysis among others.

Cryptocurrencies are a new asset class receiving significant attention from the financial and investment community. They are digital monetary and payment systems, in which value is stored in a encrypted blockchain (a decentralized and distributed network of computers that collaboratively store a ledger of transactions performed over the network) [6]. As the network of computers store duplicates of all transactions performed, any anomalies will be rapidly discovered and rejected by the network, inherently making the network exceptionally secure and trustable.

Except for Ripple (XRP), cryptocurrencies do not depend on a central authority to regulate them [7]. Most of them are forked (modified) versions of the original Bitcoin algorithm [8]. All cryptocurrencies have different rules regarding the maximum money supply, currency generation, privacy, the rate of transaction addition to the blockchain, and various proof of work mechanisms for the miners to compete against each other to secure the reward (see [9-10] for examples) [8-10].

Many investors are currently looking at cryptocurrencies as an emerging asset class (as an instrument to store value). This is demonstrated in the market capitalizations of these cryptocurrencies, which vary drastically. As of 11th of May 2017, the top five cryptocurrencies in terms of market capitalization are Bitcoin (BTC, USD 29 billion), Ethereum (ETH, USD 8 billion), Ripple (XRP, USD 6.5 billion), Litecoin (LTC, USD 1.5 billion) and NEM (XEM, USD 1 billion) [11]. This list continuously changes as an increasing number of investors invest in these new currencies. The current leader in cryptocurrency is Bitcoin, the original cryptocurrency first introduced for mass use.

Bitcoin and other cryptocurrencies have the potential to be a disruptive technology to traditional financial systems because its blockchain technology effectively removes the involvement of the banking sector and monetary authorities from financial transactions. Additionally, due to its low cost of transaction, limited supply, ability to act as a store of value against volatile fiat currencies and ability to easily transfer value across state borders, the rapid success of Bitcoin in recent years has seen Bitcoin being used increasingly in commerce and recognition of digital cryptocurrencies as an emerging asset class, value transfer and exchange method, expanding its trade to over-the-market trade and derivatives market. In 2015, it was estimated that the market capitalization of Bitcoinwas approximately USD 7 billion with USD 60 million of value changing daily [6]. Currently, the value of Bitcoin all Bitcoins in existence has significantly increased to USD $35 billionwith $1 billion in daily exchange.

In this paper, we present a neural network-based Non-Linear Autoregressive with Exogenous Input (NARX) Bitcoin price prediction model. The model uses two types of inputs. The first set of inputs were several commonly available data namely the opening, closing, minimum and maximum daily past prices. The second set of inputs consist of several Moving Average (MA) over different intervals (5, 10, 20, 50, 100, 200 days), a commonly used technical indicator for investors to estimate the direction of stock prices. These inputs were then used to train the MLP neural network to predict the next day prices for Bitcoin.

The remainder of this paper is organized as follows: Section II presents a summary of previous related works, followed by the methodology in section III. Section IV presents the results and discussions. Finally, concluding remarks are presented in section V.

2. LITERATURE REVIEW

This section summarizes several previous works related to forecasting (Section II-A) and some information regarding PSO (Section II-B) and Bitcoin (Section II-C).

2.1. Works on Financial Data Forecasting

Many works have tried to explore various prediction models for forecasting financial data. Several of them are summarized here:

In [12], a four-layer multi-agent framework was proposed to predict the quarterly price movements of the DAX German Stock Exchange index over a period of eight years. The four-layer architecture called Bat Neural Network Multi Agent System (BNNMAS) consists of several autonomous decision makers called agents. The researchers found that BNNMAS was a reliable tool for stock price prediction in long time periods based on accurate and reliable results where it has the highest value t-test which was 29.1298.

In [13],a Fuzzy Neural System (FNN) was presented to forecast the stock prices of the Vahostav company. The FNN structure consists of an input layer, fuzzy neurons and one output unit. The researchers found that the type of transfer function and the formulation of the B-spline curve will affect the accuracy of the FNN. The FNN was compared against two other models (Auto-Regressive (AR) and MLP). The FNN has shown to be marginally effective with 63.5 Root Mean Square Error (RMSE).

In [14], a Recurrent Neural Network (RNN) was used to predict the prices of several blue-chip stocks namely Citigroup, International Business Machine (IBM) and Exxon-Mobil. The RNN was proven as an excellent model with MSE of 0.131, 0.182 and 0.226 for Citigroup, IBM and Exxon-Mobil respectively over a 10-day prediction horizon.

In [15], a prediction model using a combination of data mining and ANN was used to predict the stock price of company. Their research suggested that ANN had shown good accuracy in prediction of the stock market, but the results were not sufficiently accurate. Therefore, data processing techniques were applied to check historical information with improved results compared to only ANN.

Research by [16] used machine learning techniques to predict the rise and fall of stock prices. The hybrid Support Vector Machine (SVM) model utilized both fundamental and technical analysis to forecast and make recommendations. The Boosting algorithm was used to enhance the training process. The results suggest that the proposed SVM classifiers had managed to obtain MSE of 0.467 and 0.6 respectively.

In [17], a decision tree-based algorithm was used for feature selection, and technical indicators were used in predicting the stock trends. Forecasting was performed using SVM with Radial Basis Function (RBF) kernel. The decision tree was used to select 30% from the total features for forecasting. The reduction appears to help overcome data overfitting with 950 examples used for training and prediction was done on next 50 days to validate the accuracy of prediction. The results achieved 56% accuracy on predicting the next day price trend noting that accuracy can be improved to 70% by including additional sentiment data to the prediction.

2.2. Particle Swarm Optimization (PSO)

PSO is a population-based stochastic optimization technique based on Swarm Theory (ST) and Evolutionary Computation (EC) [18]. A common analogy to help understand PSO compares it to a flock of birds in search of food. There is only one piece of food in the area being searched, which is the objective of every bird. Though not all members of the flock do not know where the food is exactly, they know roughly how far the food is from themselves as well as the relative positions of other birds in the flock. By using this information, the best strategy for individuals within the flock is to try and maintain an optimum distance between itself and the bird nearest to the food. Using this flocking strategy, eventually, all birds will land on the food. This demonstrates the underlying concept of PSO, where individuals benefit from its personal experience as well as discoveries and previous experiences of the flock.

PSO iteratively searches for solutions in the problem space by taking advantage of the cooperative and competitive behavior of simple agents called particles. The PSO algorithm search is directed by its velocity equation, $\mathrm{V_{id}}$, (Equation(1)):

$$\mathrm{V_{id}} = \mathrm{V_{id}} + \mathrm{C_1}(\mathrm{pBest} - \mathrm{X_{id}}) \times \mathrm{rand_1} + \mathrm{C_2}(\mathrm{gBest} - \mathrm{X_{id}}) \times \mathrm{rand_2} \qquad (1)$$

which modifies the particle’s position, $\mathrm{X_{id}}$:

$$\mathrm{X_{id}} = \mathrm{X_{id}} + \mathrm{V_{id}} \qquad (2)$$

where $\mathrm{V_{id}}$ = particle velocity, $\mathrm{X_{id}}$ = particle position, pBest = particle’s best fitness so far, gBest = best solution achieved by the swarm so far, $\mathrm{C_1}$ = cognition learning rate, $\mathrm{C_2}$ = social learning rate and $\mathrm{rand_1}, \mathrm{rand_2}$= random numbers between 0 and 1.

C_1 and C_2 are used to balance the influences of the swarm or individual during V_{id} update. C_1 specifically used to control the trend of particle so it favours the influence by itself (pBest) rather than that of the swarm (gBest) while for C_2, the process is vice versa. The values of C_1 and C_1 are usually set as 2.05 as these values are generally accepted [18].

2.3. Works on Bitcoin Forecasting

This section presents several related works to Bitcoin forecasting. In [19], an Artificial Neural Network (ANN)-based Bitcoin forecasting model was constructed to predicts its price over a one-day prediction horizon using its previous-day price and trading volume. Several approaches were proposed to determine the optimal architecture of the prediction ANN to output probabilities for three states which were up, margin or down.

Reference [20] explored several models (RNN, Auto-Regressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM)) to predict Bitcoin prices sourced from the Bitcoin Price Index (BPI). Training was performed using Graphics Processing Unit (GPU) due to the volume of computations required to train the models. The RMSE of the LSTM, RNN and ARIMA models were 6.87%, 5.45% and 53.47% respectively indicating that RNN was the best choice between the three models to perform Bitcoin forecasting.

In [21], a comparison between several regression models was performed for Bitcoin price prediction. The baseline, linear, SVM and ANN regression models were used to predict Bitcoin's future prices based on past data. The results indicated that the linear regression method was the most suitable for prediction. The researchers conclude that the direction of Bitcoin's price movement classification could be predicted with accuracy of approximately 55%.

3. METHODOLOGY

3.1. Data Collection

Bitcoin historical prices were collected from [22] between 12th March 2012 to 11th March 2017. The price data consists of four elements namely opening, low, high and closing price. In this model, the closing price was used as the output to the model. The closing price was chosen because it reflects all the activities of the index in the trading interval. The dataset was divided into training, validation and testing sets according to a 70%:15%:15% ratio. The dataset division was necessary to use the Early Stopping (ES) algorithm to avoid overfitting during MLP training.

3.2. Generate Moving Average Indicator

Technical indicators are predictive mathematical models attempting to estimate the future price of an asset. In this work, we used the Moving Average (MA) indicator as additional inputs to the MLP to improve its prediction. The MA indicators' lags were 5, 10, 20, 50, 100, 200 days which covers both short-term as well as long term trends.

3.3. Construct MLP NARX Model

The NARX model was realized using the MLP neural network. All programs were implemented in MATLAB version 2016b. As this MLP is used for function approximation, the hidden units use the tangent-sigmoid activation function while the output unit uses the linear activation function.

3.4. Parameter Optimization Using PSO

The NARX model construction involves the adjustment of several parameters, namely the number of MLP hidden units (H), the input lag space (n_u) and the output lag space (n_y). The PSO algorithm was used to optimize these values simultaneously to minimize the Mean Squared Error (MSE) fitness function. The PSO algorithm has several modifiable parameters, which are described in Table 1.

Table 1. PSO parameters, values and descriptions

Parameter	Value	Description
Swarm Size	50	Number of particles participating in the search process. This value needs to be sufficiently large to adequately cover the solution space. However, a value that is too high would be unfeasible as higher number of particles require higher computational power.
Maximum Iterations	500	The maximum number of iterations allowed to search for the solution. Ideally it should be high. However, a too high value would be computationally unfeasible. Therefore, the value of 500 was chosen as the swarm size was sufficiently large.

Random Seed	0	The PSO solution is highly dependent on its initial state, which is governed by the Mersenne-Twister algorithm in MATLAB. This algorithm is a pseudo-random number generator. The random seed value serves as a pointer to the random numbers generated for the experiment. Using the same initial seed would allow the particle positions to be set similarly before each optimization run, thus ensuring the experiment's repeatability over many test runs.
Dimensions	3	This parameter was set to three as there were three variables to be optimized, namely the maximum input lag (n_u), maximum output lag (n_y), and the number of MLP hidden units
Particle minimum and maximum velocities (v_{min} and v_{max})	-1 and 1	The minimum and maximum velocities define how far the particle can move at each iteration. It should cover the solution space adequately without exploring unfeasible regions. This value is tied to the minimum and maximum position ($x_{min}$and x_{max}).
Particle minimum and maximum position (x_{min} and x_{max})	0 and 1	The minimum and maximum values define the limits of the particles search at each iteration. It should cover the solution space adequately without exploring unfeasible regions.
n_u and n_y	1 to 20	The range of input and output lags explored by PSO. The value of x_{id} will be scaled according to Eq. (t-1 and t+1) to produce integer values between this range.
Number of MLP	1 to 15	Number of hidden units need to be tailored to the complexity of the project. More complex problems require more hidden units. The PSO

hidden units		algorithm was used to optimize this value.
C_1 and C_2	2.05	$C_1$is the influence of the swarm, while C_2 is the individual influence in updating particle's velocity. Acceptable general value is 2.05, balancing between individual and swarm influence.
Objective	Mean Squared Error (MSE)	The objective is usually set to 0 (no error). However, this is an ideal case and almost impossible to achieve in real-life optimization problems.

A scaling equation was used to convert the value of x_{min} and x_{max} to the desired range depending on the parameter being optimized.

$$y = (y_{max} - y_{min})\frac{x - x_{min}}{x_{max} - x_{min}} + y_{min} \quad (3)$$

where the values of y_{max} and y_{min} were set according to the ranges of the MLP hidden units, n_u and n_y.

3.5. Model Validation

Model validation is required to ensure that the model can accurately define the system's behavior. Two categories of tests are required to validate the model. The first one involves checking whether the residuals are random and uncorrelated, typically performed using autocorrelation, cross-correlation and residual histogram test. The second test measure the model fit using One-Step Ahead (OSA), MSE and R^2 analysis.

4. RESULTS AND DISCUSSION

4.1. PSO Optimization Results

Fig. 1 shows the MSE fitness versus iterations. The MSE value was initially high because the particle values have not been optimized yet. This is followed by a rapid decrease in fitness value as the particles were being optimized. The optimal solution was found during the initial iterations, and this fitness was retained until the maximum iteration. The final MSE for the optimization exercise was 206.7661. Although this value appears to be high, note that this value should be interpreted relative to the magnitude of the original data as the dataset was not preprocessed or scaled in any way. Therefore, the MSE is relatively small compared to the magnitude based on this comparison. The PSO algorithm discovered that the optimal values for n_u, n_y and H to be 10, 1 and 13 respectively.

4.2. MLP Training Performance

The MLP training plot is shown in Fig. 2. Training was performed using the Levenberg-Marquardt (LM) algorithm as the algorithm has shown superior performance in function approximation tasks [23]. The MSE was initially high but is gradually reduced as the MLP weights were updated. Training was stopped after four iterations as the ES algorithm had detected overfitting of the training set (consecutive increase in validation set training error six times (Fig. 3)).

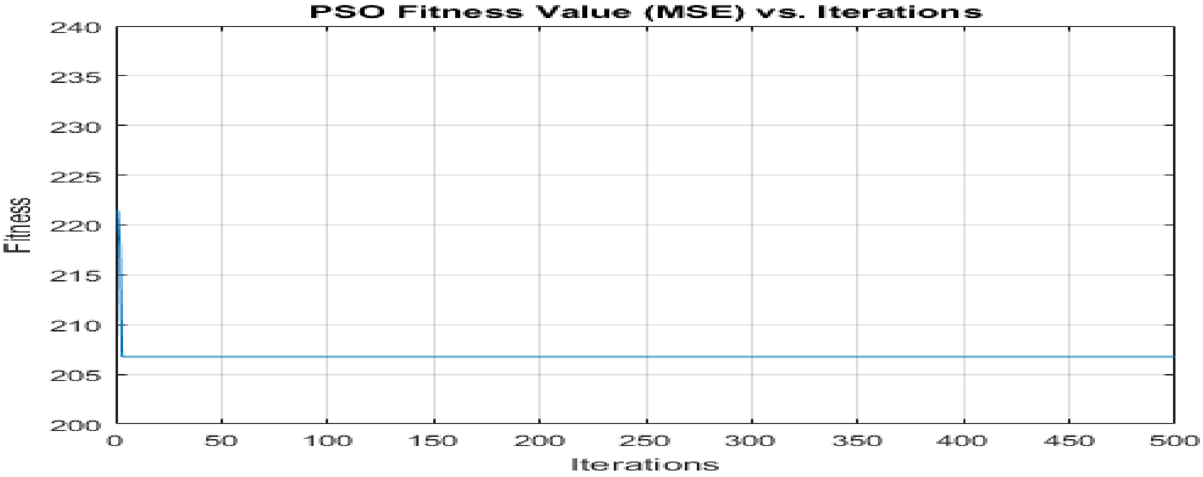

Fig.1.PSO optimization plot

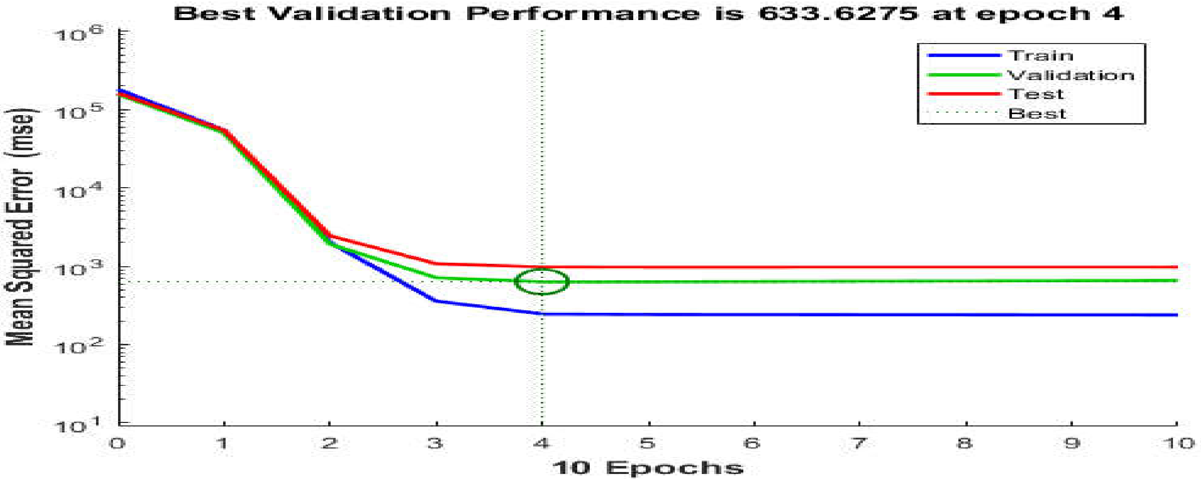

Fig.1.MLP training performance

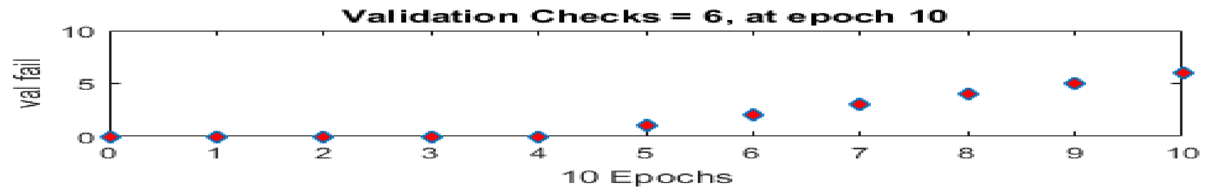

Fig.2. Training stopped after six consecutive increases in validation set error

4.3. Histogram of Residuals

Fig. 4 shows the residuals histogram, which shows the distribution of differences between the actual output and the MLP-predicted output. An important requirement for the prediction model is that the model is random as this signals that the model has effectively captured all the dynamics of the dataset, leaving only white noise residuals. The distribution of residuals is shown in Fig. 4. The residual distribution appears to follow the Gaussian bell-shaped curve, proving the residuals are randomly distributed. From this observation, we can interpret that the model is unbiased, valid and acceptable.

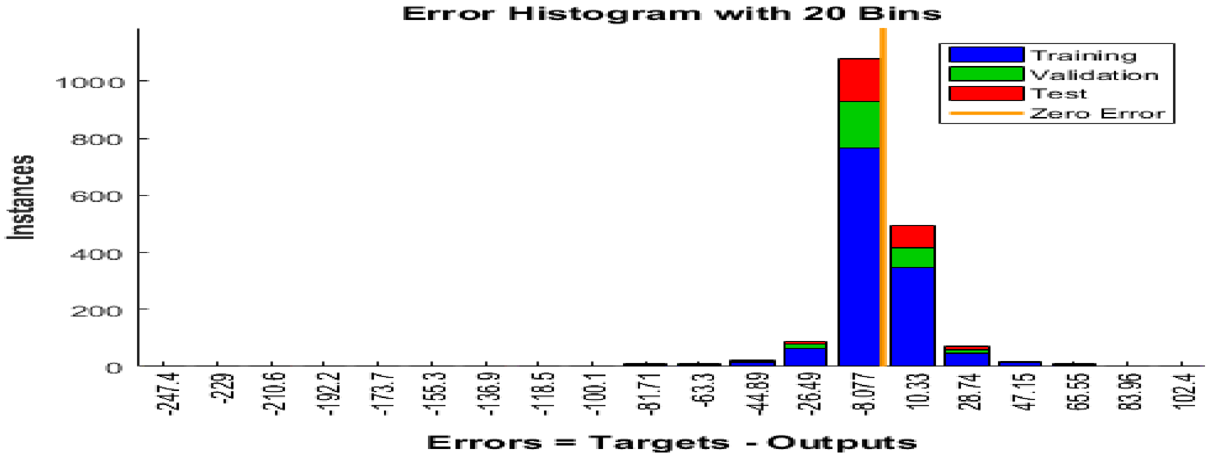

Fig.4. Histogram of MLP residuals

4.4. Model Fit using Regression Analysis

Regression analysis measures how closely the target matches the desired output. Based on the observation in Fig. 5, all datasets demonstrated desirable correlation coefficients (above 0.95), indicating a good model fit.

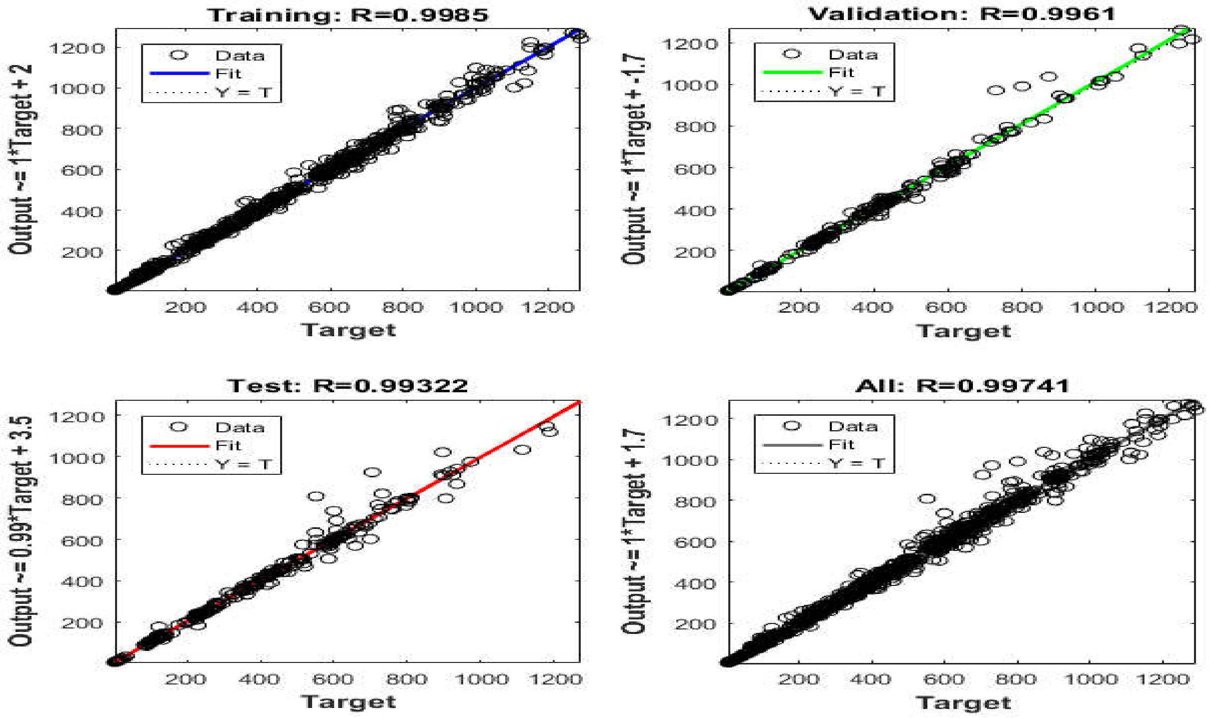

Fig.3. Regression analysis of training, validation and testing sets

4.5. One Step Ahead (OSA) Prediction

The OSA test is used to evaluate the predictive ability of the model as it tries to predict one step forward based on its past data. The OSA test results are shown in Fig. 6 and Fig. 7. A high level of overlap between actual and predicted values for both training and testing sets indicate that good model fit was achieved, thus demonstrating that the NARX model can predict future Bitcoin prices from its past trends and behaviors.

Fig.6 and Fig. 7 show the residual plots for the training set and testing set. The prediction error was smaller for the training set (range between USD 50), compared to the testing set to (range between USD 200). The reason for this is the testing set is unbiased, as it was not used during the MLP training. Therefore, the error was larger because it has not previously encountered before during the training stage. Additionally, higher residuals were observed when there were active price variations (during many peaks and valleys) as the rapid price fluctuations make the model's task of predicting the output more difficult.

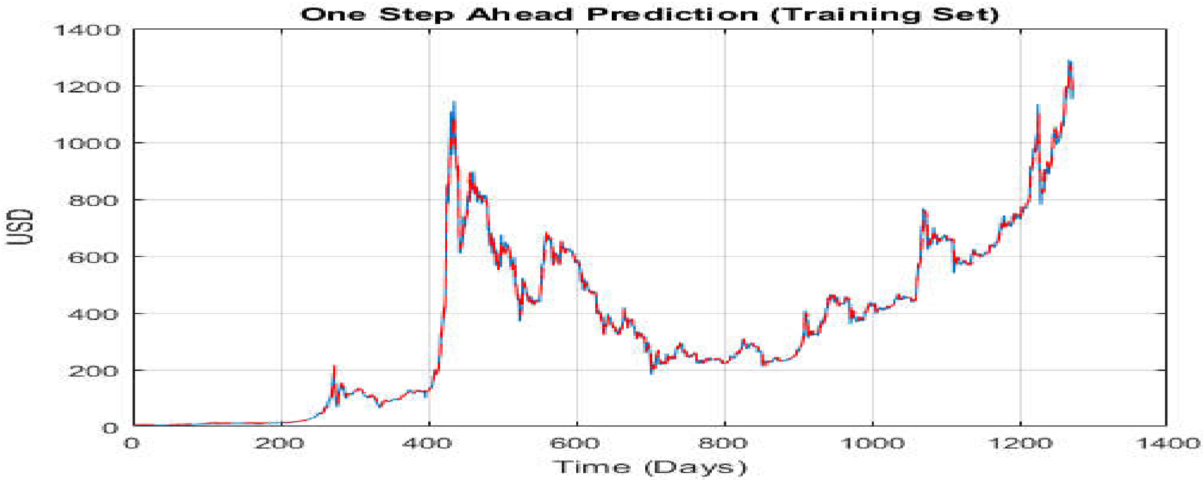

Fig.6. OSA prediction for training set

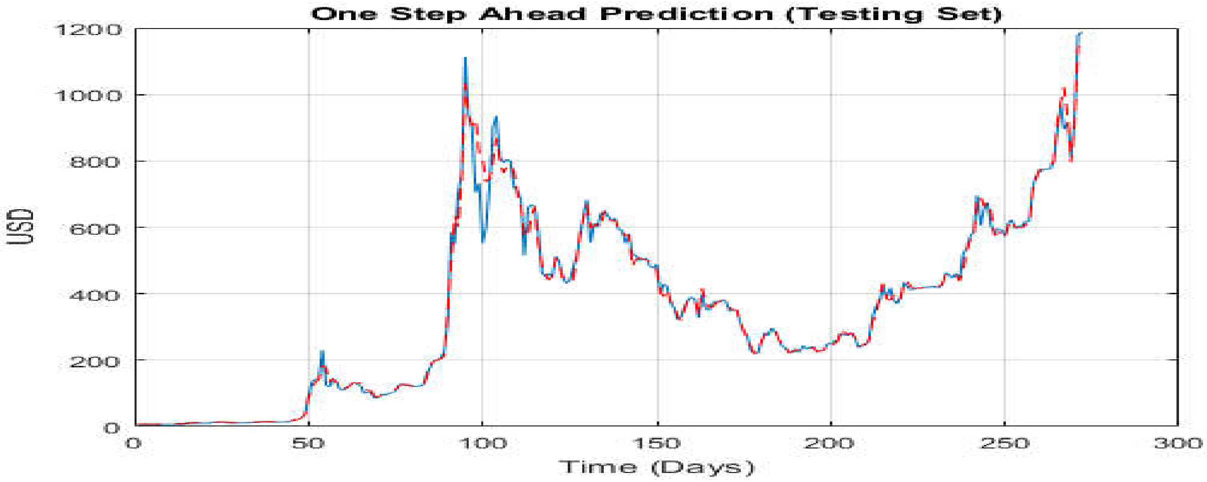

Fig.7. OSA prediction for testing set

4.6. Autocorrelation Plot

Fig. 8 shows the residual plots for the training set and testing set. For training data set, the residual plot shows that generally in the range of +1 and -1. But above 400 data points, the residual magnitude of this dataset increased to between +100 and -100. This is attributed to the period of high volatility (major swings in Bitcoin price).

The autocorrelation plots measure the similarity of the residuals with itself at different lags. This similarity is indicated by the correlation coefficients in the plot. For the autocorrelation test in Fig. 9, if a signal (in this case the residuals) is random, the similarity is highest at lag0. While for other lags, the correlation coefficient (red signal) should be in between of the 95% confidence limits (blue line in the correlation plot). The autocorrelation results for NARX model was accepted because generally, the correlation coefficient was generally between of the 95% confidence limits.

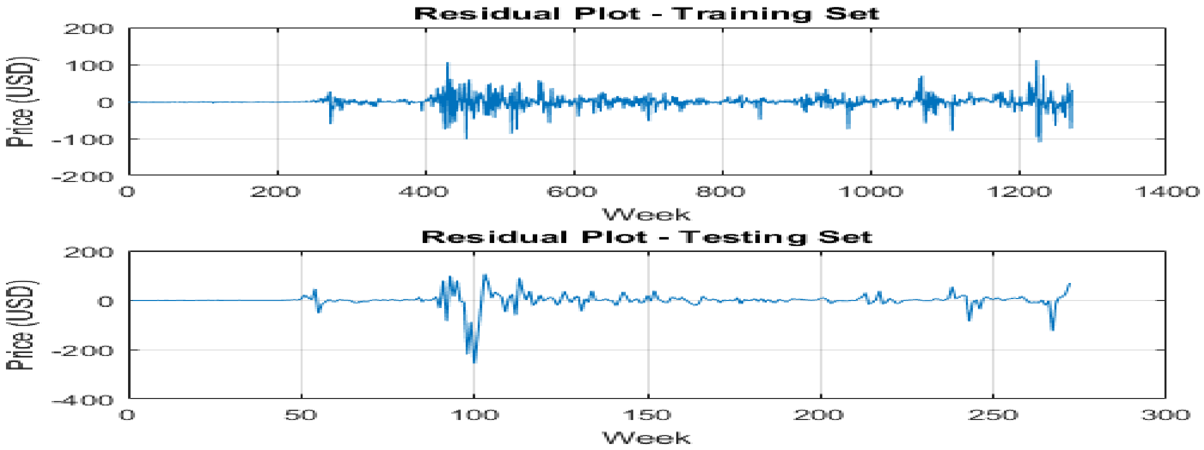

Fig.8. Residual plots for training and testing sets

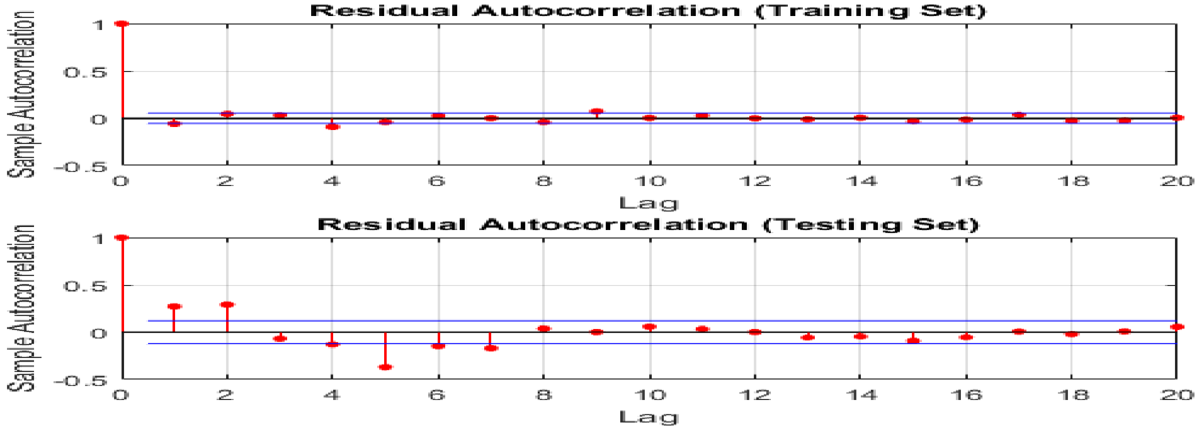

Fig.9. Autocorrelation of residuals for training and testing sets

4.7. Cross-Correlation Plot

The cross-correlation test measures the similarity of the residuals with the various input signal at different lags. The cross-correlation test results are shown in Fig. 10 (residuals vs. open price), Fig. 11 (residuals vs. high price) and Fig. 12 (residuals vs. low price).

Similar to the autocorrelation test, the cross-correlation tests show minimal coefficient violations within the 95% confidence limits. Therefore, the NARX model was accepted to be a valid model for predicting Bitcoin prices.

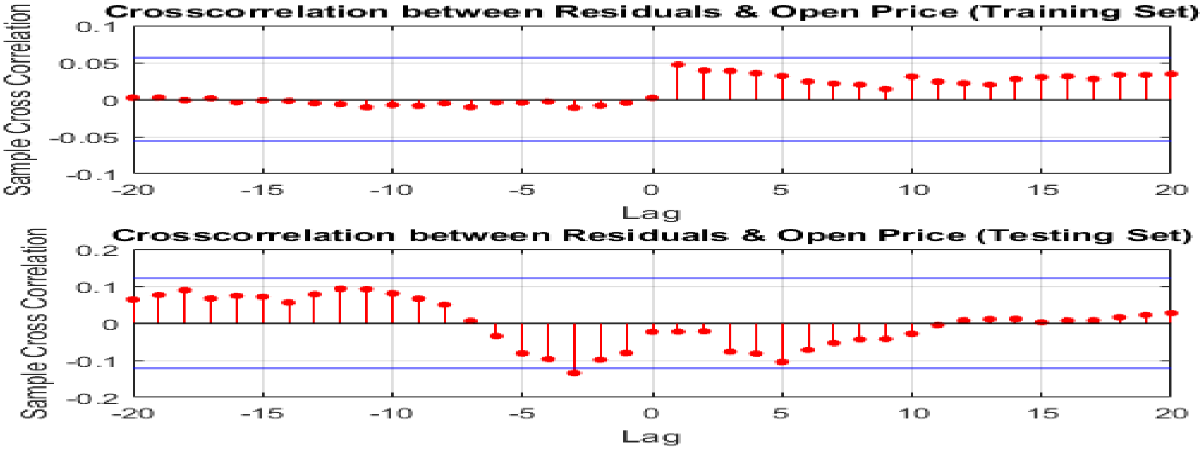

Fig.10. Cross-correlation between residuals and open price for training and testing sets

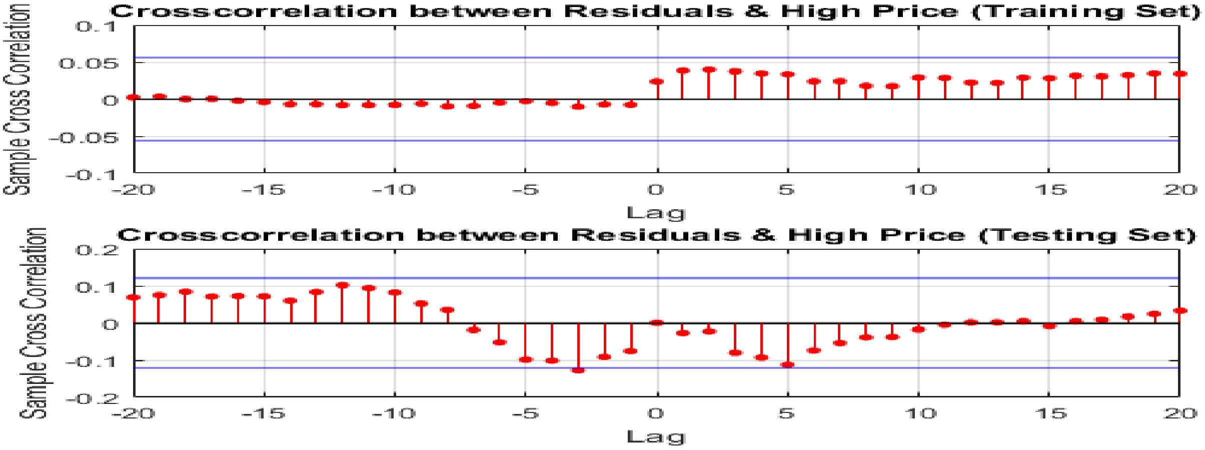

Fig.11. Cross-correlation between residuals and high price for training and testing sets

5. CONCLUSION

Cryptocurrencies are an active and interesting research area due to its potential as a substitute for fiat currency and rapid adoption in the era of internet commerce. Bitcoin is the cryptocurrency with the highest market capitalization and is the subject of this research.

A MLP-based NARX prediction model for Bitcoin price forecasting is presented in this paper. The model used commonly obtainable data together with the MA technical indicators. The PSO algorithm was used to optimize the number of hidden units, input and output lags of the NARX model.

Validation and fitting tests were performed to ensure that the model is valid and acceptable. Validation tests (autocorrelation, cross-correlation and residual histogram analysis) indicate that the model was valid, while fitting tests (OSA and regression test) indicate that the model could fit the dataset well.

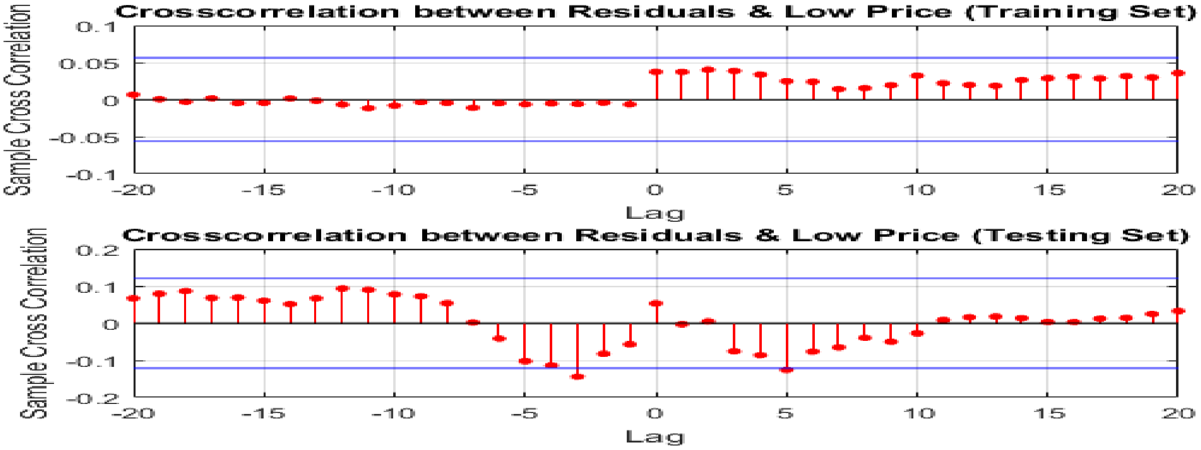

Fig.12. Cross-correlation between residuals and low price for training and testing sets

6. ACKNOWLEDGEMENTS

The authors would like to thank UniversitiTeknologiMARA and the Ministry of Higher Education Malaysia (Grant No: 600-RMI/FRGS 5/3 (0105/2016)) for the financial and infrastructure in support of this research.

7. REFERENCES

[1] Ma W, Wang Y, Dong N. Study on stock price prediction based on BP neural network. In IEEE International Conference on Emergency Management and Management Sciences, 2010, pp. 57-60

[2] Cocianu C L, Grigoryan H. An artificial neural network for data forecasting purposes. Informatica Economica, 2015, 19(2):34-45

[3] Sathe S S, Purandare S M, Pujari P D, Sawant S D. Share market prediction using artificial neural network. International Education and Research Journal, 2016, 2(3):74-75

[4] Adebiyi A A, Ayo C K, Adebiyi M O, Otokiti S O. Stock price prediction using neural network with hybridized market indicators. Journal of Emerging Trends in Computing and Information Sciences, 2012, 3(1):1-9

[5] Ariyo A A, Adewumi A O, Ayo C K. Stock price prediction using the ARIMA model. In

UKSim-AMSS 16th IEEE International Conference on Computer Modelling and Simulation (UKSim), 2014, pp. 106-112

[6] Hayes A S. Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin. Telematics and Informatics, 2016, 1-14 (article in press)

[7] Iwamura M, Kitamura Y, Matsumoto T, Saito K. Can we stabilize the price of a Cryptocurrency?: Understanding the design of Bitcoin and its potential to compete with Central Bank money.2014, http://www.ier.hit-u.ac.jp/~kitamura/PDF/P39.pdf

[8] Gandal N, Halaburda H.Competition in the cryptocurrency market.Bank of Canada Working Paper, No. 2014-33, 2014, https://www.econstor.eu/bitstream/10419/103022/1/791932281.pdf

[9] Park S, Pietrzak K, Alwen J, Fuchsbauer G, Gazi P. Spacecoin: A cryptocurrency based on proofs of space. IACR Cryptology ePrint Archive, 2015, pp. 1-26

[10] Duffield E, Hagan K. Darkcoin: Peer-to-peer crypto-currency with anonymous blockchain transactions and an improved proof-of-work system. 2014, https://pdfs.semanticscholar.org/b05d/a03086ac0b24d316bc604b25c9859df34339.pdf

[11] CoinMarketCap. Crypto-currency market capitalizations. 2014, https://coinmarketcap.com/

[12] Hafezi R, Shahrabi J, Hadavandi E. A bat-neural network multi-agent system (BNNMAS) for stock price prediction: Case study of DAX stock price. Applied Soft Computing, 2015, 29:196-210

[13] Marček D. Some intelligent approaches to stock price modelling and forecasting.Journal of Information, Control and Management Systems, 2004, 2(1):1-6

[14] Wah B W, Qian M L. Constrained formulations and algorithms for predicting stock prices by recurrent FIR neural networks. International Journal of Information Technology and Decision Making, 2006, 5(4):639-658

[15] Kshirsagar G, Chandel M, Kakade S, Amaria R. Stock market prediction using artificial neural networks. International Journal of Advanced Research in Computer Engineering and Technology, 5(5):1691-1695

[16] Shah V H. Machine learning techniques for stock prediction. 2007,

https://pdfs.semanticscholar.org/5c6d/3b9959a8556a533a58e0ac4568dfc63d6ff4.pdf

[17] Di X. Stock trend prediction with technical indicators using SVM. 2014, https://pdfs.semanticscholar.org/af39/96f74a477b649f4cf5c87645dae12b6232b9.pdf

[18] Yassin A I. Face detection using multilayer perceptrons trained on min-max features and optimized using particle swarm optimization. Master thesis, Selangor: Universiti Teknologi MARA, 2008

[19] Almeida J, Tata S, Moser A, Smit V. Bitcoin prediciton using ANN. 2015, http://s3.amazonaws.com/academia.edu.documents/39637630/Bitcoin_prediciton_using_ANN_Group_7.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1501566310&Signature=KVu8Z0%2FsHtadPadPmSzLvjs9kVs%3D&response-content-disposition=inline%3B%20filename%3DBitcoin_Stock_Prediction_Using_Artificia.pdf

[20] McNally S. Predicting the price of Bitcoin using Machine Learning. Master thesis, Dublin: National College of Ireland, 2016

[21] Greaves A, Au B. Using the Bitcoin transaction graph to predict the price of Bitcoin. 2015, http://snap.stanford.edu/class/cs224w-2015/projects_2015/Using_the_Bitcoin_Transaction_Graph_to_Predict_the_Price_of_Bitcoin.pdf

[22] Federal Reserve Board,Penny Hill Press. Bitcoin: Technical background and data analysis. South Carolina: CreateSpace Independent Publishing Platform, 2015

[23] Demuth H. B., Beale M. H., De Jess O., Hagan M. T. Neural network design. Oklahoma: Martin Hagan; 2014

How to cite this article:
Indera N I, Yassin I M, Zabidi A, Rizman Z I. Non-linear autoregressive with exogeneous input (narx) bitcoin price prediction model using pso-optimized parameters and moving average technical indicators. J. Fundam. Appl. Sci., 2017, *9(3S), 791-808*