Short Term Prediction on Bitcoin Price Using ARIMA Method

I Made Wirawan
Electrical Engineering Department
State University of Malang
Malang, Indonesia
made.wirawan.ft@um.ac.id

Triyanna Widiyaningtyas
Electrical Engineering Department
State University of Malang
Malang, Indonesia
triyannaw.ft@um.ac.id

Muchammad Maulana Hasan
Electrical Engineering Department
State University of Malang
Malang, Indonesia
has.maulana@gmail.com

Abstract— **Bitcoin is currently the most widely used digital currency. The popularity of bitcoin continues to increase and become an asset of investment. To handle with the erratic bitcoin price changes, a prediction method is needed to help its users in predicting the price in the future. By utilizing a method that is able to recognize the pattern of change in the data time series in a certain period of time can be known bitcoin price for several days ahead with a high degree of accuracy. This research uses experimental methods. Data obtained from www.coingecko.com on May 1, 2013 to June 7, 2019. Preprocessing stage includes attribute removal, stationary test, and differencing. Determination of the model candidate using correlogram method. The predictions are done using the Autoregressive Integrated Moving Average (ARIMA) method, which is capable of generating high accuracy in short-term predictions. Evaluating the prediction results using Mean Absolute Percentage Error (MAPE). The results showed that ARIMA (4,1,4) models resulted in predictions with the smallest MAPE, 0.87 for the next one-day prediction and 5.98 for the next seven days. Thus the ARIMA (4,1,4) model is feasible to be used as a predictive method of Bitcoin for one to seven days ahead.**

Keywords —Bitcoin, Prediction, ARIMA, MAPE.

I. INTRODUCTION

The rapid development of technology makes all human activities not separated from the role of technology, no exception to the financial sector. Digital currencies or commonly called cryptocurrencies are increasingly popular in use, one example is Bitcoin. Bitcoin first appeared in 2008 and began to be widely used in the year 2012 [1]. Bitcoin itself is a digital currency that is open source and for storage media is not centralized but is stored using a peer-to-peer network and uses cryptography to secure its data security [2]. In addition, Bitcoin also allows ownership without identity so that the confidentiality of the owner of an account is secured [3].

Unlike conventional currencies, the development of Bitcoin prices is no supervision and is not controlled due to its undecentralized nature [4]. The inflation control is implemented in bitcoin transactions but limited only and can be found by all parties. This leads to a very unstable price of the bitcoin exchange rate. In a matter of minutes the price of Bitcoin can change several times. Bitcoin users must be observant in overseeing any price changes in order to be benefited instead of being harmed by mistake when making a transaction when the value is down.

The unstable price of Bitcoin can be anticipated by making a prediction or forecasting the price of Bitcoin in an upcoming time period [5]. With predictions, users can determine when to make a transaction. The main thing to note in prediction is the accuracy level of the predictive method done [6]. Therefore, it takes a method that is able to overcome problems that can’t be solved manually so as to make the right decision so that the parties concerned are not to suffer losses.

Some research on doing bitcoin price predictions has been done. Those are the predictions of Bitcoin prices using Recurrent Neural Network (RNN) [7]. The study resulted in the best average accuracy gained by 98.76% in the training data and 97.46% in test data, with the best number of input pattern parameters is 5, number of Epoch 1000, learning rate value of 0.001 and number of hidden units 50. Further predictions of Bitcoin price changes using neural networks [8]. The research is using the Perceptron and Recurrent Neural Network Multiplayer architectures. The results show that long-term predictions have better results than short-term predictions, Multilayer Perceptron outperform Recurrent Neural Networks with 81.3% accuracy, 81% precision and 94.7% recall.

Some studies using the ARIMA method for predictions have also been performed. Such as Stock Price Prediction of PT. BRI, TBK Using ARIMA Method" [9]. The study uses maximum and minimum data on stocks from 3 January 2011 to 20 October 2014 to predict the stock price of November 2014. The result of the ARIMA model for the maximum share price is ARIMA (2,1,3) and the minimum share price is ARIMA (2,1,3). Further use of the ARIMA method to predict the number of passenger trains in Malang City [10]. The Model of ARIMA (0,1,7) was able to predict the number of passenger trains from Malang City with MAE value of 1.37409, RMSE of 1.17721, and accuracy value of 98.01%.

ARIMA is a predictive method that fully ignores independent variables in making forecasting, making it suitable for interconnected statistical data (dependent) and has some assumptions that must be fulfilled like autocorrelation, or seasonal trends [11]. Some of the advantages of this algorithm are able to produce high level of accuracy in short term prediction, reliable and efficient in predicting time series financial data, able to process large-scale data and analyze random data situation [12]. The ARIMA method was chosen because it was able to predict large time series data and was able to produce good short-term predictions. So using the ARIMA method is expected that the research is able to produce a good price prediction of bitcoin and be a material consideration of users in the face of price changes on Bitcoin.

978-1-7281-3832-9/19/$31.00 ©2019 IEEE

II. Research Methods

The research is divided into 5 main phases, namely data collection, preprocessing, model candidate determination, model testing and evaluation, and best model determination.

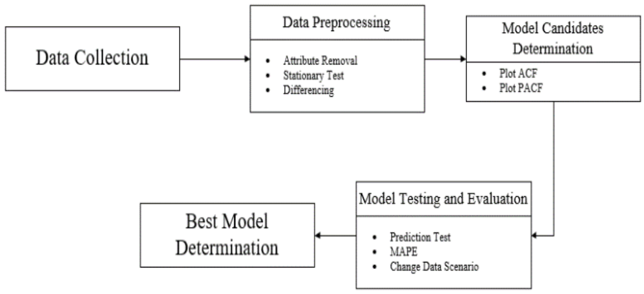

Fig. 1. Design Research

A. Data Collection

The Data used in this study was obtained from the site www.coingecko.com. Coingecko is a site that provides a wide range of information about cryptocurrencies such as Bitcoin, Ethereum, Ripple, litecoin and hundreds of other cryptocurrencies. The Data provided is updated at all times so that the end assured. Coingecko provides data export features so that the price history of each cryptocurrency can be easily obtained.

The exported Dataset from the Coingecko site is a CSV file in which there are 2227 data. The Dataset contains a history of Bitcoin prices from 1 May 2013 to 7 June 2019. The Dataset has 4 attributes: snapped_at, price, market_cap, and total_volume.

B. Preprocessing

At this phase, the dataset that has been obtained will be processed to be used in the research. The data preprocessing consist of attribute removal, stationary test, and differencing.

Attribute removal is done to select or remove unneeded attributes in the prediction process. These attributes are not used because they will not affect the final outcome or will even reduce the accuracy level. The selection of attributes used is tailored to the data needs of the ARIMA method.

A stationary test is performed to determine whether the data is stationary or not. Stationary test can be done by looking directly at the graphic plot of the dataset used, if the chart is on a straight line then the data is already stationary. The second way is to do the ACF plot on the data, if there is a significant change in its value, then the data is stationary.

A time series that is not stationary should be converted into stationary data by doing a differencing process. The meaning of differencing is to calculate the change or difference in the observation value [13]. The value of difference earned is checked again whether stationary or not. If it is not stationary then it is done differencing again. Differencing processes can be performed by reducing the value of one period by a previous period value, or by the following equation.

$$Y_{d_t} = Y_t - Y_{t-1} \tag{1}$$

A higher order difference is calculated in the same way. For example, the second sequence difference (d = 2) is only expanded to include the second lag of the series, the similarities are as follows.

$$Y_{d2_t} = Y_{d_t} - Y_{d_{t-1}} \tag{2}$$

C. ARIMA

The Autoregressive Integrated Moving Average (ARIMA) is a method developed by George Box and Gwilyn Jenkins in 1970 and is commonly referred to as the Box-Jenskins method [14]. The ARIMA method fully ignores independent variables in making forecasting, making it suitable for interconnected statistical data (dependent) and has some assumptions that must be fulfilled such as autocorrelation, trend, or Seasonal. The ARIMA method is capable of predicting historical data with the influence of data that is difficult to understand technically and has a high degree of accuracy in short-term forecasting and is capable of dealing with seasonal data fluctuations. In determining the method of ARIMA that fits the need to be tested on Asumsi-asumsinya, because one good method for a case is not necessarily suitable for another case. The ARIMA method is divided into 4 groups, namely Autoregressive (AR), Moving Average (MA), Autoregressive Moving Average (ARMA) and Autoregressive Integrated Moving Average (ARIMA) [15][16].

The AR Model was first introduced by yule in 1926 and then developed again by Walker in 1932. This Model has the assumption that data is now affected by previous period data. It is called autoregressive because on this model it is remodeled against the previous values of the variable itself. The AR method is used to determine the order value of the coefficient p that indicates the dependency of a value with the previous closest value [17].

The general form of AR Model with order p (AR (p)) or ARIMA model (p,0, 0) is stated as follows.

$$X_t = \mu + \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} + e_t \tag{3}$$

The MA Model was first introduced by slutzky in 1973. The MA method is used to determine the value of the order coefficient q which explains the variable movement of the previous residual value [18].

The general form of MA model with order q (MA (q)) or ARIMA model (0,0, q) is stated as follows.

$$X_t = \mu + e_t - \emptyset_1 e_{t-1} - \emptyset_2 e_{t-2} + \cdots + \emptyset_q e_{t-q} \tag{4}$$

The ARMA model is a composite of the AR and MA models. Assuming that the current period data is affected by previous period data and the forced value of the previous period.

Common forms of the model of the AR and MA or ARIMA processes (p, 0, q) are stated as follows.

$$X_t = \mu + \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} + e_t - \emptyset_1 e_{t-1} - \emptyset_2 e_{t-2} - \cdots - \emptyset_q e_{t-q} \tag{5}$$

The ARIMA Model assumes that the data used must be stationary which means the average variation of the data in question is constant. Non stationary data must be changed first in order to be stationary through a differencing process. ARIMA method is a statistical perspective method that is represented by three parameters, the first of the data AR process of the past period is taken and maintained later in the Integrated process (I) makes the data become to facilitate the predictive process. The common form of the ARIMA model (p, d, q) is stated as follows.

$$X_t = \mu + X_{t-1} + X_{t-d} + \cdots + \emptyset_1 X_{t-1} + \emptyset_2 X_{t-2} + \cdots + \emptyset_p X_{t-p} + e_t - \emptyset_1 e_{t-1} - \emptyset_2 e_{t-2} - \cdots - \emptyset_q e_{t-q} \quad (6)$$

D. Model Candidate Determination

The ARIMA Model has three orders, namely p, d, q. The determination of the order of ARIMA candidates is done using a Correlogram method that can be analyzed through the plot Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). The ACF Plot is used to describe the correlation between data time series that is adjacent to time lag. The ACF plot is used to determine the candidate's value in the order q. While the PACF plot is used to measure the level of data when the influence of time lag is considered separate. The PACF Plot is used to determine the value of candidates in the order p. As for the value of the order d candidate is determined by the number of differencing process done to turn the data into stationary.

Determination of order seen from the plot results ACF and PACF by analyzing the presence of dies down and cuts off. The data is called dies down when the value of data correlation in the plot decreases slowly approaching the value of 0. While the cuts off when the correlation value decreases significantly exceeding the value of α = 0.05 or less than α = -0.05. Candidates taken based on the cut off only apply to the 1st to 10th lag, if more than that means the data requires a differencing process again. The ACF and PACF plot creation process can be performed using the ACF () and PACF () functions of the package forecast in RStudio. For the identification and determination of order candidates based on ACF and PACF plot conditions can be seen in Table I.

TABLE I. ACF AND PACF PLOT CONDITIONS

No	ACF and PACF Plot conditions	Candidate Model
1	Real ACF at the 1st lag, 2,..., q and *cuts off* on lag-q PACF *dies down*	ARIMA (0, d, q)
2	ACF *dies down* Real PACF on lag-p and *cuts off* after lag-p	ARIMA (p, d, 0)
3	Real ACF on lag-q and *cuts off* after lag-q Real PACF on lag-p and *cuts off* after lag-p	ARIMA (0, d, q) if ACF *cuts off* sharper, ARIMA (p, d, 0) if PACF *cuts off* Sharper
4	There is no real autocorrelation on the ACF and PACF plots	ARIMA (0, d, 0)
5	ACF *dies down* PACF *dies down*	ARIMA (p, d, q)

E. Testing Model for Prediction and Evaluation

After getting some candidates the next step model is testing the model. Testing is conducted on each candidate that has been determined. The testing process is divided into two phases, which is to predict and calculate the predicted error rate. Predictions are made to get the bitcoin price one to seven days ahead. After you get the predicted result, the error rate of each prediction is calculated using MAPE from the first day to the seventh day.

Testing was conducted with three data scenarios. The initial dataset with 2227 data will be divided into training datasets and test datasets. The data sharing scenario can be seen in Table II. Each scenario requires seven test data because the predicted result is only up to seven days ahead.

TABLE II. DATA CHANGE SCENARIOS

Scenario	Training Data	Test Data
1	1 May 2013 - 31 March 2019 (2159 Data)	1 April 2019 - 7 April 2019
2	1 May 2013 - 30 April 2019 (2189 Data)	1 May 2019 - 7 May 2019
3	1 May 2013 - 31 May 2019 (2220 Data)	1 June 2019 - 7 June 2019

The evaluation phase for testing accuracy prediction results on this study using Mean Absolute Percent Error (MAPE) [16]. MAPE is an alternative method of evaluating predictive techniques used to measure the accuracy of the forecasting of a model [17]. MAPE is the average of the overall percentage of errors (difference) between the actual data and the predicted data. A low MAPE value indicates that the resulting value is approaching its actual value. Here is the equation of the MAPE.

$$MAPE = \frac{1}{n}\sum\left(\frac{f_t - y_t}{f_t}\right) * 100 \quad (7)$$

where:

f_t = actual value in T period
y_t = T Period prediction value
n = amount of observation data

F. Best Model Determination

Once the test has been successfully conducted against all model candidates, the next step is to determine the best model. The best models are models that have a good performance in predicting the price of bitcoins for one to seven days ahead with a small error rate and high accuracy. Performance calculations are performed by calculating the value of MAPE in each predicted result, from the first to the seventh day predictions. Models that produce the smallest MAPE value in three scenarios are deemed to have successfully made predictions with the best results among other models.

III. RESULT AND DISCUSSIONS

A. Preprocessing

At this stage the first one is to remove the delete attribute that is not used in the prediction process. Of the four attributes on the dataset, only two attributes can be used, namely the snapped_at and price attributes. The Market_cap and Total_volume attributes are not used in the predictive process. The process of deleting an attribute is performed on the import data in the RStudio application by passing an unselected attribute in the attribute selection step.

Next change the data format in the Snapped_at attribute to simplify the process of importing data in RStudio application. The initial data Format of yyyy-mm-dd hh-mm-ss is changed to yyyy-mm-dd. This format change is done because it only requires data retrieval date, while the data retrieval time of each day is the same as at 00.00.00 so no need to Used. Datasets after the process of changing attributes and data formats can be seen in Table III.

TABLE III. DATASETS AFTER THE ATTRIBUTE REMOVAL PROCESS

snapped_at	price
2014-01-01	767.74
2014-01-02	772.53
2014-01-03	825.47
2014-01-04	849.14
2014-01-05	919.41
2014-01-06	936.38
2014-01-07	826.50
2014-01-08	838.32
2014-01-09	853.29
2014-01-10	863.30

After that, the stationary test can be done in two ways, namely by looking at the graphic plot of the original data or viewing the graphic plot of the ACF data. The ACF chart Plot can be seen in Fig. 2 and shows a significant value in the initial lag and then shrinks gradually. From both tests it can be ensured that the data is still not stationary.

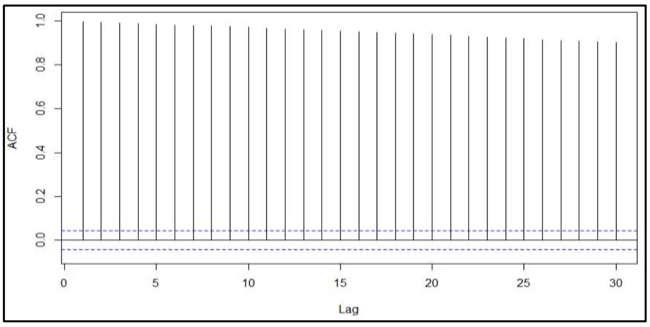

Fig. 2. Plot ACF Dataset

The next step performs a differencing process to transform the data to be stationary. The differencing process is performed using the help of the diff() function from time series package in RStudio. Firstly the differencing process is done using 1 stage and the result can be seen in Fig. 3 and Fig. 4.

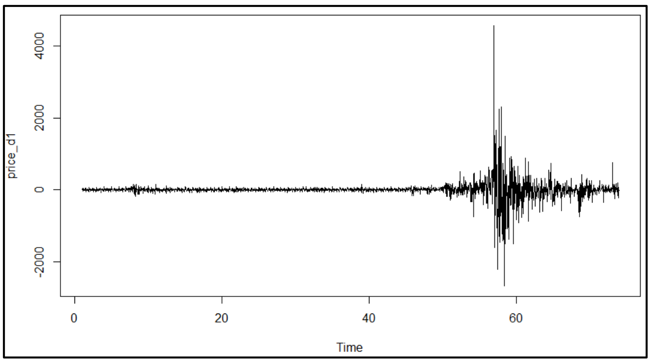

Fig. 3. Plot of Data after Differencing

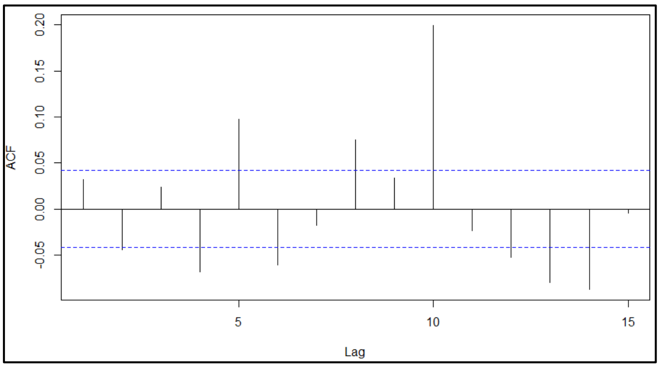

Fig. 4.Plot ACF Data after Differencing

From the data graph in Fig. 3 and the ACF plot in Fig. 4 the data has been changed to stationary. Visible from the data graph that is straight at a value of 0 in the middle and the ACF plot also indicates the change in the value of lag by cuts off or dies down.

B. *Model Candidate Determination*

The candidates p and q order can be determined by looking at the lag value on the ACF and PACF plots. To perform the ACF plotting using the ACF() function whereas for the PACF plotting it uses the PACF() function of the package forecast. the ACF and PACF plotting Result values can be seen in Table IV.

TABLE IV. THE VALUE OF LAG ACF AND PACF

Lag	ACF value	PACF value
1	0.0311671	0.0311671
2	-0.0408177	-0.0418297
3	0.0169148	0.0195972
4	-0.0653285	-0.0684481
5	0.0958295	0.1026839
6	-0.0585199	-0.0735526
7	-0.0200340	-0.0023926
8	0.0795752	0.0666964
9	0.0284593	0.0382968
10	0.1979718	0.1887868

Based on the results of ACF plot there are cuts off at some lag whose value exceeds the limit of 0.05, namely at lag 4, 5, 6, and 8. While the results of PACF plot shows the values on the lag 4, 5, 6, and 8 experienced cuts off exceeding the 0.05 limit. From these results and referring to Table III then lag 4, 5, 6, and 8 will be candidate for order p and q on ARIMA model.

Order d value is 1, because it has done 1 differencing process to make the dataset used to be stationary. From some candidates p, d, q that have been mentioned if combined, will produce the candidate ARIMA model (4, 1, 4), (4, 1, 5), (4, 1, 6), (4, 1, 8), (5, 1, 4), (5, 1, 5), (5, 1, 6), (5, 1, 8), (6, 1, 4), (6, 1, 5), (6, 1, 6), (6, 1, 8), (8, 1, 4), (8, 1, 5), (8, 1.6), and (8, 1, 8). Those 16 models will have some AR and MA term on each model. The next step to determine which model is best is to test the model for forecasting.

C. Testing Models for Predictions

Testing conducted with three predefined scenarios, viewable in Table II. Each scenario uses the same stages and processes. The first stage is to test the model to make a prediction, and then the predicted result is a calculated error rate using MAPE. All of testing is performed using RStudio program. The predictive process is done using (6) and assisted by the function of ARIMA () from the forecast package. As for the MAPE calculation process use (7) and assisted by the MAPE () function of MLmetrics package.

The first test phase is to test the model for doing Bitcoin price predictions. The test example was done on the ARIMA model (4, 1, 4) for the first scenario. The function used to perform the prediction is ARIMA (). The results can be seen in Fig. 5. This model produces four coefficients AR and four MA coefficients. The value of the coefficient is then used to predict the next period by using (6).

```
> arima(price, order = c(4,1,4))

Call:
arima(x = price, order = c(4, 1, 4))

Coefficients:
          ar1     ar2     ar3      ar4      ma1      ma2     ma3     ma4
      -0.0024  0.8596  0.0456  -0.8615  -0.0704  -0.9976  0.0374  0.9212
s.e.   0.0221  0.0161  0.0259   0.0254   0.0157   0.0149  0.0310  0.0244

sigma^2 estimated as 53353:  log likelihood = -14808.32,  aic = 29634.65
```

Fig. 5. Value of ARIMA Model coefficient (4, 1, 4) Scenario 1

The results of prediction using the ARIMA model (4, 1, 4) for the seven future periods can be seen in Fig. 6. To display the predicted results using the forecast () function. The output of the function is the prediction value, high, and low of the prediction.

```
> forecast(arima(price, order = c(4,1,4)), h = 7)
         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
6.911020       3976.480 3680.464 4272.496 3523.762 4429.198
6.913758       3879.083 3475.410 4282.756 3261.718 4496.448
6.916496       3817.553 3351.130 4283.976 3104.220 4530.885
6.919233       3833.673 3309.221 4358.124 3031.593 4635.752
6.921971       3886.032 3316.759 4455.306 3015.403 4756.661
6.924709       3980.863 3361.724 4600.002 3033.971 4927.754
6.927447       4079.386 3406.700 4752.071 3050.602 5108.169
```

Fig. 6. ARIMA Model Prediction (4, 1, 4) Result Scenario 1

Next calculate the predicted *error rate* with the actual price using MAPE. To calculate MAPE using the MAPE () function and the result can be seen in Fig. 7.

```
> for (x in 1:7){print(MAPE(forecast(arima(price, order = c(4,1,4)),
 h = 7)$mean[1:x], btc_data_test$price[1:x])*100)}
[1] 3.095922
[1] 4.770567
[1] 10.34223
[1] 13.41747
[1] 14.86945
[1] 15.82791
[1] 16.2939
```

Fig. 7. Model MAPE value ARIMA (4, 1, 4) Scenario 1

Using the same testing method, then applied to all other model candidates.

D. Evaluation

After testing on each model and getting the prediction result for the next seven days, the following step is to calculate the predicted error rate using MAPE. Calculation of MAPE is done with (7). The calculation process is done using the help of MAPE () function. After the value of MAPE for the first and seventh day predictions, the results are averaged. The final result obtained for each model can be seen on Tables V.

TABLE V. MAPE RESULTS

Model	Scenario 1	Scenario 2	Scenario 3
(4,1,4)	11.23	**3.24**	**2.92**
(4,1,5)	11.18	3.58	3.23
(4,1,6)	7.30	4.47	3.62
(4,1,8)	10.10	4.58	4.10
(5,1,4)	11.14	3.71	3.34
(5,1,5)	6.90	4.91	3.81
(5,1,6)	**6.86**	4.92	3.79
(5,1,8)	9.23	5.33	4.74
(6,1,4)	7.31	4.44	3.60
(6,1,5)	6.86	4.98	4.13
(6,1,6)	7.33	4.56	3.91
(6,1,8)	9.86	4.76	3.66
(8,1,4)	10.27	4.77	4.32
(8,1,5)	10.08	4.99	4.41
(8,1,6)	9.93	4.88	3.80
(8,1,8)	9.58	5.00	3.86

Test results against 16 model candidates with three scenarios indicate each scenario resulted in a changing value of each model's error rate. In the first scenario of the ARIMA model (5,1,6) generates the lowest error rate value with an average of 6.86. In the second scenario of ARIMA models (4,1,4) generates the lowest error rate value with an average of 3.24. The third model of ARIMA (4,1,4) is the lowest value of the error rate with an average of 2.92.

The Model ARIMA (4,1,4) was chosen as the best model to perform Bitcoin predictions up to seven days in the future because it has managed to get good results with a low error rate in both the second and third scenario tests. This Model uses four AR coefficients, four MA coefficients, and one differencing stage. The values of AR and MA coefficients are also below 1 all, indicating that the condition of the stationarity has been met. During the second scenario testing, this model could result in predictions of the same change pattern as the actual price, but still has a price gap. As for the third scenario testing, this model was able to produce predictions for the first and second days that approached the actual price.

The smaller the value of the d and q parameters, the risk of overfitting the model is smaller. It is possible for an AR term and an MA term to cancel each other's effects, so if a mixed AR-MA model seems to fit the data, so a model with fewer AR MA parameter is better.

IV. Conclusions

ARIMA can perform bitcoin price predictions for one to seven days ahead with good results. Some models have been tested and the model ARIMA (4,1,4) can predict the price of Bitcoin with the best level of accuracy. MAPE generated 0.87 for first day predictions (1 June 2019) and 5,98 for the Seventh Day Prediction (June 7, 2019).

ARIMA's performance is better if it is used for short-term predictions, especially for predictions of the two future periods. The more periods are predicted the lower the level of accuracy. ARIMA can be used for bitcoin price prediction but with significant price difference but is able to predict price change patterns in the seven future periods. The irregular/patterned bitcoin price change characteristics make it difficult to determine the corresponding ARIMA model. One good model when doing predictions on a single period range is not necessarily good for the next period range. Of the three Tests with different data models generates a changing error rate value.

References

[1] H. Halaburda and N. Gandal, "Competition in the Cryptocurrency Market," 2014.

[2] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *J. Gen. Philos. Sci.*, 2008.

[3] E. Androulaki, G. O. Karame, M. Roeschlin, T. Scherer, and S. Capkun, "Evaluating user privacy in Bitcoin," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7859 LNCS, pp. 34–51, 2013.

[4] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *Proceedings - IEEE Symposium on Security and Privacy*, 2013, pp. 397–411.

[5] Rob J Hyndman, "Forecasting: Principles and Practice," no. September, 2014.

[6] J. D. Cryer and K.-S. Chan, *Time Series Analysis With Applications in R*, vol. 20, no. 2. 2008.

[7] R. A. Juanda, Jondri, and A. A. Rohmawati, "Bitcoin Price Prediction by Using Recurrent Neural Network," *e-Proceeding Eng.*, vol. 5, no. 2, pp. 3682–3690, 2018.

[8] R. Albariqi, "Bitcoin Price Change Prediction Using Artificial Neural Network," 2018.

[9] G. S. Lilipaly, D. Hatidja, and J. S. Kekenusa, "Stake Price Prediction Using ARIMA Method at PT. BRI, TBK.," *J. Ilm. Sains*, vol. 14, pp. 61–66, 2014.

[10] A. Qonita, Muladi, and T. Widiyaningtyas, "The use of ARIMA Method to Predict the Number of Train Passengers in Malang," 2018.

[11] A. Qonita, A. G. Pertiwi, and T. Widiyaningtyas, "Prediction of Rupiah Against Us Dollar by Using ARIMA," *Int. Conf. Electr. Eng. Comput. Sci. Informatics*, vol. 4, no. September, pp. 746–750, 2017.

[12] J. C. Paul and S. Hoque, "Selection of best ARIMA Model for Forecasting Average Daily Share Price Index of Pharmaceutical Companies in Bangladesh: A Case Study on Square Pharmaceutical Ltd.," *Glob. J. Manag. Bus. Res.*, vol. 13, no. 3, pp. 15–26, 2013.

[13] N. A. Bakar and S. Rosbi, "Autoregressive Integrated Moving Average (ARIMA) Model for Forecasting Cryptocurrency Exchange Rate in High Volatility Environment: A New Insight of Bitcoin Transaction," *Int. J. Adv. Eng. Res. Sci.*, vol. 4, no. 11, pp. 130–137, 2017.

[14] E. B. Dagum, "The X-II-ARIMA seasonal adjustment method," 2005.

[15] A. Hendranata, "ARIMA (Autoregressive Integrated Moving Average)," 2003.

[16] Y. S. Lee and L. I. Tong, "Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming," *Knowledge-Based Syst.*, vol. 24, no. 1, pp. 66–72, Feb. 2011.

[17] Hillmer, S. Craig, and George C. Tiao, "An ARIMA-Model-Based Approach to Seasonal Adjustment," vol. 10, no. 1, pp. 5–24, 2017.

[18] S. E. Said and D. A. Dickey, "Testing for unit roots in autoregressive-moving average models of unknown order." pp. 599–607, 1984.

[19] A. de Myttenaere, B. Golden, B. Le Grand, and F. Rossi, "Mean Absolute Percentage Error for regression models," *Neurocomputing*, vol. 192, pp. 38–48, 2016.

[20] M. V. Shcherbakov, A. Brebels, N. L. Shcherbakova, A. P. Tyukov, T. A. Janovsky, and V. A. evich Kamaev, "A survey of forecast error measures," *World Appl. Sci. J.*, vol. 24, no. 24, pp. 171–176, 2013.