Available online at http://scik.org
Commun. Math. Biol. Neurosci. 2022, 2022:100
https://doi.org/10.28919/cmbn/7706
ISSN: 2052-2541

THE EXTENSION OF MOORE-PENROSE GENERALIZED INVERSE FOR EXTREME LEARNING MACHINE IN FORECASTING USD/IDR EXCHANGE RATE AS IMPACT OF COVID-19

RESTU ARISANTI*, SYELA NORIKA SIMBOLON, RESA SEPTIANI PONTOH

Department of Statistics, Padjadjaran University, Indonesia

Abstract: Since the beginning of the COVID-19 pandemic, the Indonesian economy has undergone changes which have caused the Rupiah exchange rate against the US Dollar to increase and weaken. Throughout March to September 2020, the Rupiah exchange rate depreciated by 2.75% - 4.57% with a range of Rp. 14,000 – Rp. 16,575 per US Dollar. Depreciation has a negative impact on the Indonesian economy because it makes product prices relatively cheaper for other countries. Therefore, it is necessary to do a forecast to determine the exchange rate of the Rupiah against the US Dollar in the future. Forecasting is an activity carried out using existing data to predict something in the future. In forecasting USD/IDR Exchange Rate using the Extreme Learning Machine method. This method does not require parametric assumptions and has a faster learning speed feedforward by determining the weights and biases on the network randomly. The ELM formulation leads to solving a system of linear equations in terms of unknown weights connecting the hidden layer to the output layer. The solution of this general system of linear equations is obtained using the Moore-Penrose Pseudo Inverse. The results of data analysis resulted in the optimum ELM network architecture (17-49-1), namely 17 neurons in the input layer, 49 neurons in the hidden layer, and 1 neuron in the output

*Corresponding author
E-mail address: r.arisanti@unpad.ac.id
Received August 31, 2022

layer. The network is obtained from MSE which produces the smallest error value, which is 0.000338 on training data and 0.000139 on testing data. With this network, the results of forecasting USD/IDR Exchange Rate with MAPE are 0.2383951%. Forecasting results show that the Rupiah exchange rate has appreciation and is quite stable as expected by the government in strengthening the Indonesian economy during the COVID-19 pandemic.

Keywords: USD/IDR exchange rate; COVID-19; forecasting; extreme learning machine.

2010 AMS Subject Classification: 92B20.

1. INTRODUCTION

Currency exchange rate is a comparison between the value of a country's currency with other countries. The difference in the exchange rate of a country's currency is in principle determined by the amount of demand and supply of that currency. The exchange rate is used as one of the macroeconomic indicators in measuring the stability of a country's economy because there is a balance between demand and supply that occurs in the market, given its large influence on the current account balance as well as for other macroeconomic variables.

The rupiah exchange rate against foreign currencies, especially the US Dollar is one of the important indicators in analyzing the Indonesian economy. US Dollar is used as the global currency in making international payments that apply between countries in the world. Exchange rate movements have become a serious concern for the Government and Bank Indonesia as the monetary authority to monitor and control them, especially about factors affecting the rupiah exchange rate. The exchange rate is not set by the central bank, but the market, so the exchange rate can change at any time according to market mechanisms. Therefore, predictions of future currency exchange rates are needed to determine future economic policies.

The exchange rate of the rupiah against the US dollar can strengthen or weaken at any time. The strengthening and weakening of the exchange rate can be caused by several factors, such as the amount of demand for goods and services, inflation rates, interest rates, market expectations and Central Bank intervention which have a significant impact on exchange rate fluctuations [1]. Fluctuations in the exchange rate will be a risk for investors, entrepreneurs, and banking circles in international transactions, so it is necessary to forecast the rupiah exchange rate against the US Dollar to reduce the risks that may arise. The size of the rupiah exchange rate against the dollar currency needs to be monitored regularly to ensure the economy of a country is in a stable condition or vice versa [2]. If the rupiah exchange rate against the dollar depreciates continuously, it can have an impact on the amount of state debt that must be paid. Therefore, a stable economy is needed to avoid this.

Indonesia is a developing country where the rupiah exchange rate tends to fluctuate. The COVID-19 pandemic, which has hit almost the entire world, has caused uncertainty in the economic sector, especially the Indonesian economy. This caused the Rupiah depreciated by 2.75% - 4.57% with a range of Rp. 14,000 – Rp. 16,575 per US Dollar. The weakest condition of the Rupiah was on March 23, 2020, where the Rupiah exchange rate was Rp.16,575. The Rupiah exchange rate against the US Dollar began to correct again in early June to the end of September with a range of Rp13,870-14,900/US Dollar. The Rupiah Exchange Rate is estimated to be in a vulnerable position during the COVID-19 pandemic, because of uncertainty in the global market that triggers negative sentiment towards investors to channel their investment funds to other countries compared to Indonesia [3].

Because of the importance of maintaining the stability of the IDR exchange rate against the USD for the sake of the domestic economy, we need a way to forecast the IDR exchange rate against the USD so that stakeholders and policy makers can move faster in making decisions. Forecasting is an attempt to predict the state of a particular object in the future by using past historical data [4]. Forecasting the rupiah exchange rate against the dollar will also be useful for investors because the rupiah exchange rate is used by investors as an indicator that affects activities in global markets that have an impact on investors' profits and losses in investing activities [5].

Forecasting the IDR Exchange Rate against USD is carried out using the Extreme Learning Machine method that can be applied to various types of data patterns. This method can capture the data pattern well in the training process and the generalization performance is fast with the formulation process in linear equations using Moore Penrose. This method was introduced by

Huang to overcome the weaknesses in conventional SLFNs algorithms which have a long enough training performance due to an iterative process [6]. Therefore, in this study, the ELM method will be applied to predict the IDR exchange rate against the USD.

2. PRELIMINARIES

In this research, the data used is historical data of daily exchange rate of the Rupiah against the US Dollar from January 2019 to June 2022 which is derived from investment website (www.investing.com). The data used is univariate time series data and analyzed with the Extreme Learning Machine method using R software.

Extreme Learning Machine (ELM) was introduced by Huang in 2004 as a feedforward artificial network with a single hidden layer [7]. The ELM method was created to overcome the limitations of conventional SLFNs algorithms such as the Backpropagation algorithm which is quite long in the training process. The advantage of this method are the determination of the input weight parameters and biases randomly, then the formulation uses a general linear equation to obtain the output weight with Norm Least Squares Solution and Moore-Penrose so that ELM has a fast-learning speed and can produce good generalization performance[8]. With the characteristic of Moore Penrose, theoretically it can minimize training errors so that ELM is faster in the training process and gets good results[9].

The ELM model is mathematically simpler and more effective than the feed-forward artificial network. For N arbitrary distinct samples $(\mathbf{x}_\mathrm{i}, \mathbf{t}_\mathrm{i})$, where $\mathbf{x}_\mathrm{i} = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbf{R}^n$ and $\mathbf{t}_\mathrm{i} = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in \mathbf{R}^m$, standard SLFNs with $\widetilde{N}$ hidden nodes with activation function $g(x)$ *infinitely differentiable* can approximate these N distinct samples with zero error are modelled as [6]

$$\sum_{i=1}^{\widetilde{N}} \beta_i g_i(\mathbf{x}_j) = \sum_{i=1}^{\widetilde{N}} \beta_i g(\mathbf{w}_\mathrm{i}.\mathbf{x}_j + b_i) = \boldsymbol{t}_j \qquad ,j = 1,2, \dots, N \tag{1}$$

Equation (1) can be compactly as

$$\mathbf{H}\beta = \mathbf{T} \tag{2}$$

where

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1.\mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}.\mathbf{x}_1 + b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_1.\mathbf{x}_N + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}.\mathbf{x}_N + b_{\tilde{N}}) \end{bmatrix}_{N\times\tilde{N}}, \ \beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_n^T \end{bmatrix}_{\tilde{N}\times m}, \text{dan } \boldsymbol{T} = \begin{bmatrix} \boldsymbol{t}_1^T \\ \vdots \\ \boldsymbol{t}_N^T \end{bmatrix}_{N\times m}$$

In simple terms, to train ELM on a certain weight and bias value can be described as finding the least squares solution of the linear system of equation (2). However, many cases occur in the number of hidden nodes that are smaller than the number of training data samples so that the H matrix is a non-square matrix and cannot be inverted. The solution to this problem is to use the Moore-Penrose Generalized Inverse of the H matrix. The smallest norm least squares solution of the above linear system is [6]

$$\hat{\beta} = \mathbf{H}^{+}.\mathbf{T} \tag{3}$$

Where

$$\mathbf{H}^{+} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}} \tag{4}$$

Moore-Penrose Generalized Inverse ($\mathbf{H}^{+}$) is the hidden layer output matrix of the neural network to calculate the output weight. Moore-Penrose is a singular inverse matrix of order $m \times n$. If the inverse of a common matrix is the inverse of a square matrix and is non-singular (the determinant is not zero), then a Moore - Penrose inverse exists for every matrix, whether it is a singular or a non-square matrix. The properties of Moore-Penrose in a matrix are as follows:

1. $\mathbf{HGH} = \mathbf{H}$
2. $\mathbf{GHG} = \mathbf{G}$
3. $(\mathbf{HG})^{\mathrm{T}} = \mathbf{HG}$
4. $(\mathbf{GH})^{\mathrm{T}} = \mathbf{GH}$

If these four properties are satisfied, then G is called the Moore-Penrose inverse of H with the notation $\mathbf{H}^{+}$ [6]. Based on the theorem, $\hat{\beta} = \mathbf{H}^{+}.\mathbf{T}$ is a special solution of least-squares for the system of general linear equations $\mathbf{H}\beta = \mathbf{T}$, which means that the smallest training error can be achieved by:

$$\left\|\boldsymbol{H}\hat{\beta} - \boldsymbol{T}\right\| = \|\mathbf{H}\mathbf{H}^{+}\mathbf{T} - \mathbf{T}\| = \min_{\beta}\|\mathbf{H}\beta - \mathbf{T}\| \tag{5}$$

The following are the steps in the analysis of the ELM method [10]:

1. Split the data in the proportion of 80% for training data and 20% for testing data
2. Pre-processing data by normalizing the data using the Min Max Normalization by following equation:

$$X_i' = \frac{x_i - X_{min}}{X_{max} - X_{min}}(b - a) + a \tag{6}$$

   where X_i' is the value of the data after normalization, x_i is the original value of the data, X_{min} is the minimum value of the entire original data, X_{max} is the maximum value of the entire original data, a is the minimum range value, b is the maximum range value.
3. Forming an ELM network architecture which consists of an input layer, a hidden layer, and an output layer. Determination of input layer neurons is based on the significant lag in the PACF plot of the variables to be used as input [11]. The hidden layer is used to receive and send signals to the neural network. According to [11] one hidden layer is enough to solve forecasting problems. The number of hidden layers consists of 1 layer which contains several neurons with a binary sigmoid activation function. The number of neurons in the hidden layer is determined by trial and error to get the number of neurons that produces the expected error value. The output layer consists of only 1 neuron because the data used in this study is univariate time series data and the prediction process only produces one real value.
4. Implementation of training and testing process. The training process using the ELM method consists of three steps, namely initializing the input and bias weights, calculating the output of the hidden layer, and calculating the output weight. Initialization of input weights and bias of each input unit is obtained from randomization. Then, calculate the number of outputs generated from each hidden unit. The output weight (β) will be calculated by a general linear equation which has a special solution using Moore Penrose ($\mathbf{H}^{+}$) and after that it is activated using a Binary Sigmoid activation function. Each unit of output is compared with its target value to determine the size of the error (MSE) during the

training process. If the MSE value is still relatively large, then the process is repeated by changing the number of hidden layer units by a certain amount. After getting a relatively small MSE value with an efficient number of hidden layer units, the weights and biases obtained from the training data can be used for data testing and forecasting. The steps in the testing process are the same as the training process, but the weight calculation is no longer carried out because all weights are taken from the results of the previous training

5. Evaluation of the forecasting result model using MAPE

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\% \tag{7}$$

6. Forecasting by changing the value (data denormalization) into the original form with the following equation:

$$X_i = \left[\frac{X_i' - a}{(b-a)}\right](X_{max} - X_{min}) + X_{min} \tag{8}$$

3. Main Results

3.1 Descriptive Analysis

The first step that needs to be done is to perform a descriptive statistical analysis on the USD/IDR rate variable to explore the data. The results of the descriptive statistics are shown in the Table 1.

Table 1. Descriptive statistics of USD/IDR Exchange Rate

Variable	Minimum	Maximum	Mean	Q1	Median	Q3
Rupiah (IDR) to USD rate	13,572.50	16,575.00	14,329.17	14095	14,265.00	14447.5

Table 1. shows the minimum, maximum, mean, quantile 1, median, and quantile 3 values of the USD/IDR Exchange Rate variable. It shows that the lowest value of the rupiah exchange rate against the US Dollar is Rp. 13,572.50 and the highest value reached Rp. 16,575.00.

Figure 1 shows a line chart of the USD/IDR Exchange Rate for the period January 2019 to June 2022. Based on Figure 1, it shows a large fluctuation and a sharp increase since the COVID-19 pandemic. Throughout March to September 2020, the rupiah exchange rate depreciated by 2.75% - 4.57% with a range of Rp. 14,000 – Rp. 16,575.00 per US Dollar. This certainly has a negative impact on the Indonesian economy, so it is necessary to observe further on the movement of the Rupiah exchange rate so that it remains stable. Therefore, in this research, forecasting will be carried out for the next few days using Extreme Learning Machine.

Figure 1. Line Chart of USD/IDR Exchange Rate

3.2 Formation of The ELM Model

In this study, the actual data will be normalized using the min max normalization technique. The data will be split into 2 parts with the proportion of 80% training data and 20% testing data. Next, form the most optimum network architecture. The network architecture in the Extreme Learning Machine (ELM) model consists of an input layer, a hidden layer, and an output layer, also known as a multilayer perceptron. Each layer needs to be determined first the number of neurons in each layer. In the input layer, the number of neurons is determined based on the significant lag in the PACF plot. The PACF plot in Figure 2. shows that the significant lag is at lag 1-17. Thus, the number of neurons in the input layer is 17 inputs, consisting of $X_1, X_2, \cdots, X_{17}$.

In the hidden layer, the determination of the number of neurons is based on trial and error, namely the forward approach method by conducting experiments starting from the smallest number of hidden neurons to the largest number. The number of neurons analyzed will start from 1 neuron to 50 neurons and the optimum number of neurons will be selected based on the minimum Mean Square Error (MSE) value. The number of output neurons used is 1 neuron because prediction result with only one real value.

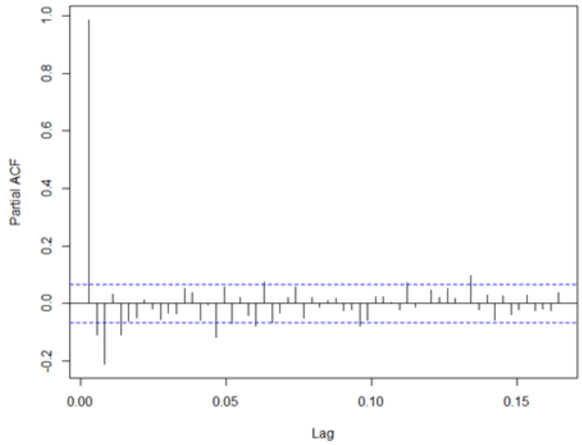


Figure 2. Plot PACF of USD/IDR Exchange Rate

Table 2. ELM model network architecture of USD/IDR

Input Neurons	17 neurons
Hidden Layer	1 layer
Hidden Neurons	1-50 neurons (trial and error)
Output Neurons	1 neuron

After forming the network architecture, the next step is the training and testing process using the binary sigmoid activation function. The steps of the training process begin with initializing the input weights and bias randomly with values between 0-1. Next, perform a feedforward procedure to calculate the value of the output layer. Each input unit receives and forwards the signal to the hidden layer. Then, the hidden layer unit sums the weighted input signals and is subjected to an activation function to get the output of the hidden layer.

Table 3. The Error Sizes of ELM Model of USD/IDR Exchange Rate

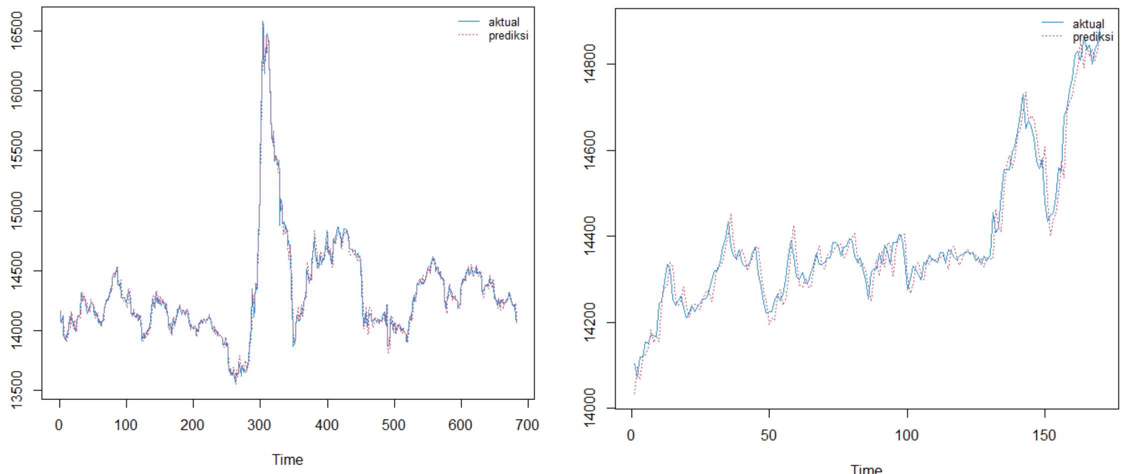
Hidden Neurons	**MSE Training**	**MSE Testing**	**Hidden Neurons**	**MSE Training**	**MSE Testing**
1	0.024182938	0.004372029	26	0.000488	0.000154
2	0.010010867	0.001240445	27	0.000497	0.000177
3	0.003314739	0.000717251	28	0.000459	0.000176
4	0.003341321	0.000679042	29	0.000493	0.000147
5	0.002671482	0.000534214	30	0.000424	0.000158

Hidden Neurons	MSE Training	MSE Testing	Hidden Neurons	MSE Training	MSE Testing
6	0.003419992	0.000918539	31	0.000419	0.000143
7	0.003628031	0.000869228	32	0.000424	0.000185
8	0.001776317	0.000645344	33	0.000451	0.000143
9	0.000831078	0.00022918	34	0.000483	0.000167
10	0.001111569	0.000303235	35	0.000437	0.000149
11	0.001052157	0.000240214	36	0.000433	0.000143
12	0.00081658	0.000226345	37	0.000454	0.000147
13	0.001476143	0.000385231	38	0.000401	0.000145
14	0.00059879	0.000300598	39	0.00039	0.000145
15	0.001160719	0.000269835	40	0.000387	0.000141
16	0.000641362	0.000196541	41	0.000377	0.000151
17	0.001028823	0.000323316	42	0.000403	0.000139
18	0.000628118	0.00016343	43	0.000374	0.000133
19	0.000619983	0.000178136	44	0.000384	0.000133
20	0.000587643	0.000149714	45	0.000385	0.000142
21	0.000501486	0.000174898	46	0.000368	0.000151
22	0.000556888	0.000156648	47	0.000355	0.000129
23	0.000494584	0.000160846	48	0.000373	0.000139
24	0.000504984	0.000143712	49	0.000338	0.000139
25	0.000498742	0.000151766	50	0.000349	0.000136

The output is collected in a matrix H with the order $N \times \tilde{N}$ which will be used to find ouput weight using Moore Penrose. The output weight is calculated and activated

by the activation function. Iterations carried out in the feedforward steps will stop when the expected error value is at its minimum by looking at the MSE value in the network that is formed. The MSE value is used to measure the performance of the network, namely how well the network can recognize patterns. In the network formed, there are 2 types of MSE, namely MSE training and MSE testing. The optimum network is the network that has the smallest MSE value both in the training and testing process. However, if the MSE training and testing values are different, then the network will be selected based on the MSE in training because the training process is the formation of the model to be used. The training process will stop when the network shows that the MSE value has increased.

Figure 3. Graphs of Actual and Predicted Values for (a) USD/IDR data training (b) USD/IDR data



(a) (b)

testing

Based on Table 3, it shows that the MSE training and testing value is the minimum with 49 hidden neurons so that the optimum ELM network architecture that is formed can be written with the ELM model (17-49-1). In the training and testing process, the error value obtained from the forecasting results for each training data and testing data is then compared with the target value. The following is a comparison plot of the forecast results for training and testing data

The blue line in Figure 3 represents the target data in the form of original data from the USD/IDR Exchange Rate, while the red dotted line shows the prediction results (network output results) of the USD/IDR Exchange Rate using the ELM network (17-49-1). Based on Figure 3, it can be seen that the difference between the actual data and the forecast results is quite small because the forecast data pattern follows the actual movement pattern of the USD/IDR Exchange Rate data. The visualization of the network architecture of the ELM model (17 - 49 - 1) is shown in Figure 4.

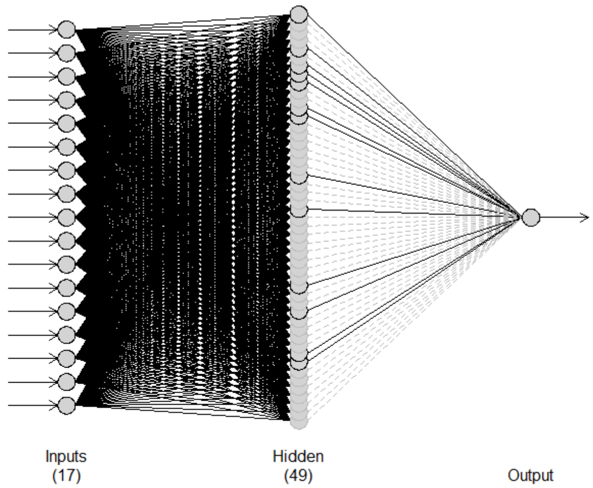


Figure 4. The network architecture of the ELM model (19-47-1)

The next step is to forecast using the network architecture in Figure 4 for the next 10 days. In this study, forecasting is done using actual data that is complete or not divided. The MAPE value obtained from the network is 0.2383951%, which means that the accuracy of the forecasting results obtained can be said to be very good. Forecasting results are shown in table 3

Table 3. Forecast results of USD/IDR Exchange Rate

Date	Exchange Rate	
01/07/2022	Rp	14,162.92
02/07/2022	Rp	14,128.77
03/07/2022	Rp	14,072.04
04/07/2022	Rp	14,067.00
05/07/2022	Rp	14,103.28
06/07/2022	Rp	14,109.35
07/07/2022	Rp	13,992.26
08/07/2022	Rp	13,923.22
09/07/2022	Rp	13,958.86
10/07/2022	Rp	13,949.72

From the forecasting results shown in table 3, the USD/IDR Exchange Rate is predicted to be around Rp. 13,000 – Rp. 14,000 per US Dollar. Forecasting results show that the Rupiah exchange rate has appreciation and is quite stable as expected by the government in strengthening the Indonesian economy during the COVID-19 pandemic.

In this study, it can be concluded that forecasting exchange rate of the Rupiah against the US Dollar can use the Extreme Learning Machine (ELM) method. The most optimum Extreme Learning Machine network architecture based on the minimum MSE value is the ELM network (17-49-1) which uses input lag from data with 17 neurons in the input layer, 49 neurons in the hidden layer and 1 neuron in the output layer. The network provides the minimum MSE value, which is 0.000338 for MSE training and 0.000139 for MSE testing. This study uses a binary sigmoid activation function and conducts trial and error in finding hidden neurons until it reaches the minimum error value. The ELM artificial neural network model with the architecture described above can be used properly to predict the rupiah exchange rate against the dollar exchange rate with an error (MAPE) of 0.2383951%. This MAPE value indicates that the network built has very good accuracy. So, it can be concluded that the network that has been built can be used to predict exchange rate of the Rupiah against the US Dollar.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] A. Sartono, Manajemen Keuangan Teori dan Aplikasi, Edisi Empat. BPFE, Yogyakarta, 2001.

[2] A.B. Fatahillah, Faktor-faktor yang Mempengaruhi Fluktuasi Nilai Tukar Rupiah Terhadap Dollar Amerika Serikat, Jurnal Berkala Ilmiah Efisiensi, 16 (2016), 524-535.

[3] A.F. Putri, E. Suprapto, Pengaruh Pandemi Covid-19 dan Nilai Tukar Rupiah terhadap Pergerakan Indeks Harga Saham Gabungan (IHSG) di Bursa Efek Indonesia (BEI), Thesis, Universitas Brawijaya, 2021.

[4] P. Subagyo, Forecasting (Konsep dan Aplikasi), BPFE, Yogyakarta, 1999.

[5] R.R. Elvierayani, Peramalan Nilai Tukar (Kurs) Rupiah Terhadap Dolar Tahun 2017 dengan Menggunakan Metode Arima Box-Jenkins, Prosiding SI MaNIs, 1 (2017), 253–261.

[6] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: Theory and applications, Neurocomputing. 70 (2006), 489–501. https://doi.org/10.1016/j.neucom.2005.12.126.

[7] G.B. Huang, Q.Y. Zhu, C.K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541), IEEE, Budapest, Hungary, 2004: pp. 985–990. https://doi.org/10.1109/IJCNN.2004.1380068.

[8] S. Ding, H. Zhao, Y. Zhang, X. Xu, R. Nie, Extreme learning machine: algorithm, theory and applications, Artif. Intell. Rev. 44 (2015), 103–115. https://doi.org/10.1007/s10462-013-9405-z.

[9] S. Lu, X. Wang, G. Zhang, X. Zhou, Effective algorithms of the Moore-Penrose inverse matrices for extreme learning machine, Intell. Data Anal. 19 (2015), 743–760. https://doi.org/10.3233/IDA-150743.

[10] R. Singh, S. Balasundaram, Application of extreme learning machine method for time series analysis, Int. J. Intell. Technol. 2 (2007), 361–367.

[11] G. Zhang, B. Eddy Patuwo, M. Y. Hu, Forecasting with artificial neural networks:: The state of the art, Int. J. Forecast. 14 (1998), 35–62. https://doi.org/10.1016/s0169-2070(97)00044-7.