Tadulako Social Science and Humaniora Journal
Vol 2, No. 2, July 2022, pp. 22-34

APPLICATION OF THE *EXTREME LEARNING MACHINE* (ELM) METHOD IN PREDICTING THE COMBINED STOCK PRICE INDEX (IHSG) IN INDONESIA

Nur Azizah Janad [a*] , Junaidi b , Iman Setiawan [c]
Statistics Study Program, Faculty of Mathematics and Natural Sciences, Tadulako University
Corresponding Author*: nura78249@gmail.com

Abstract

The stock market is one area that continues to attract the attention of investors and financial researchers. This research explores the application of the Extreme Learning Machine (ELM) method to predict the Composite Stock Price Index (IHSG) in Indonesia. ELM is known for its fast learning capabilities with minimal prerequisite network architecture. In this research, three types of activation functions, namely Sigmoid, ReLU, and Tanh, are applied to ELM to compare their performance in predicting IHSG. Monthly IHSG data is used for model training and testing. Data preprocessing steps, such as dividing the data into Training and Test sets, are applied before feeding it into the model. Model performance was evaluated using Root Mean Square Error (RMSE) and compared for each activation function. The research results show that each activation function has a different impact on the IHSG prediction performance. In this research, the ReLU activation function showed the best performance in predicting IHSG compared to other activation functions, with a Root Mean Square Error (RMSE) of 1 x 10-16. These results show that the model's predictive performance in estimating actual values is very good.

Keywords: ***Extreme Learning Machine***; ***Composite Stock Price Index***; **Activation Function; Prediction**

1. INTRODUCTION

Current technological advances have a big impact on people's lives. It is easier for the community to carry out activities that are beneficial from an economic and employment perspective. For example, from a financial perspective, people can easily make money using the internet, one of which is investing. Investment is the activity of postponing consumption allocated to productive assets for a certain period of time in the hope of gaining profits in the future. Investment in the form of shares is commonplace to hear and do. Many Indonesian and foreign investors invest and trade shares in Indonesia. Shares are proof of ownership of a company. Buying shares means that shareholders who are part of the company's ownership have the right to receive company profits in the form of dividends (Halimi & Kusuma, 2018).

There are various types of shares in Indonesia, one of which is the Composite Stock Price Index (IHSG), which in English is also called the Indonesian Composite Index, ICI, or IDX Composite. The composite stock index (IHSG) is an indicator that can show stock price movements, which functions as a market indicator. In other words, it represents the movement of indicators that show whether the market is active or weak (Basit, 2020). The composite stock index (IHSG) is one of the factors that reflects the performance of the capital market, regardless of whether the capital market experiences an uptrend (rise) or a downtrend (fall) (Widodo, 2018). To analyze the possibility (prediction) of an increase or decrease in stock prices in the future can be done in various ways, one of which is by developing artificial intelligence techniques, in this case, the most widely used for the above purposes is using Artificial Neural Networks.

Artificial neural networks (ANN) are an information processing system that has characteristics resembling biological neural networks (JSB). Artificial Neural Networks were created as a generalization of the mathematical model of human understanding (human cognition) (Sudarsono, 2016). Artificial neural networks work to process information based on how the human brain works by carrying out training and testing processes through changing weights (synapses), and recognition based on past data. Past data is studied, resulting in decisions based on data that has never been studied (future) (Yanti et al., 2018). The artificial neural network structure consists of three layers, namely the input layer, the hidden layer, and the output layer. Each layer is given a weight that will transform the input value into an output value. Each layer consists of several neurons and these neurons are connected to other neurons in nearby layers.

In this research, the method used to predict is an Extreme Learning Machine (ELM). Extreme Learning Machine (ELM) is a method that works on the concept of a single-layer feedforward network (SLFN). This method was created to overcome the weaknesses of the feed-forward artificial neural network method, especially in the learning rate process (Alfiyatin et al., 2018). ELM has interesting and important features that conflict with common gradient-based learning algorithms for feed-forward neural networks. One of the features is that ELM has a very fast learning speed. In replications reported in the literature, the ELM learning step can be completed in seconds in many applications (Mardiana et al., 2020).

Previous research conducted by (Masriyah et al., 2022) on predicting the composite stock price index (IHSG) used a dynamic model of Newton's cooling law. In this research, three dynamic models were applied from the results of modifications to Newton's cooling law which will be used to predict the composite stock price index (IHSG), namely the Price Reversion Model, Price Reversion-Quasi Logistics Model, and Velocity Reversion Model. Based on the validation results, it was found that the best model for predicting the composite stock price index (IHSG) was the Price Reversion Model with a MAPE value of 8.4159. Research on extreme learning machine (ELM) has also been carried out, one of which is by (Riandri et al., 2018) with the title implementation of the extreme learning machine (ELM) algorithm in predicting geomagnetic storm activity. The results of his research stated that by using the extreme learning machine (ELM) method, with or without regularization, the prediction results were equivalent to a prediction accuracy of 57.80822%. Based on the description provided, in this research the extreme learning machine (ELM) method will be used to predict the Composite Stock Price Index (IHSG) in Indonesia.

2. RESEARCH METHODS

2.1 Data Sources and Research Variables

The data used in this research is secondary data obtained from the investing.com website. The variable used is the composite stock price index.

2.2 Analysis Method

Data analysis in this research uses the Extreme Learning Machine (ELM) method with the help of R software. The stages of analysis carried out are as follows:

1. Enter Composite Stock Price Index (IHSG) data.
2. Determine the lag value using PACF.
3. Normalize the data.
4. Carrying out training and testing processes on Composite Stock Price Index (IHSG) data.
5. Model the training data using the Extreme Learning Machine (ELM) method.
6. Carry out the prediction process using the Extreme Learning Machine (ELM) with sigmoid, ReLU, and tanh activation functions.
7. Evaluate the performance of the Extreme Learning Machine (ELM) method using RMSE.
8. Conclusion

3. RESULT D DISCUSSION

3.1 Data Exploration

Data exploration aims to gain an understanding of the Composite Stock Price Index (IHSG) data patterns. In this research, the data analyzed is Composite Stock Price Index (IHSG) data from January 2019 to August 2023 (Appendix 1). Figure 4.1 depicts the Composite Stock Price Index (IHSG) data pattern in the time period January 2019 to August 2023.

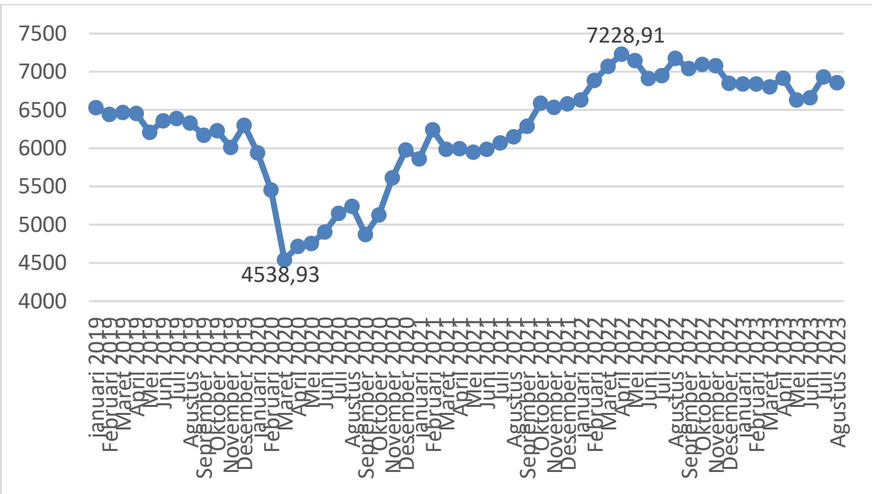

Figure 1. Composite Stock Price Index (IHSG) *Time Series Plot*

Figure 1 shows that during the period January 2019 to August 2023, the IHSG experienced fluctuations every month. In March 2020, IHSG experienced a quite drastic decline, namely 4,538.93 compared to declines in other months. However, in the following month, namely April 2022, there was a quite drastic increase, namely 7,228.91 compared to the increase in other months. Based on the results of the analysis that has been carried out, it can be seen that the data JCI shows a high level of volatility or experiences frequent changes in a short time span. This can be seen from the IHSG fluctuations depicted in the graph in Figure 1.

- Determining the Amount of Lag Using Pacf

In this research, a PACF plot was used to determine the number of lags to be used. The PACF plot used was obtained from R software. The following are the results of the Pacf plot obtained from R software:

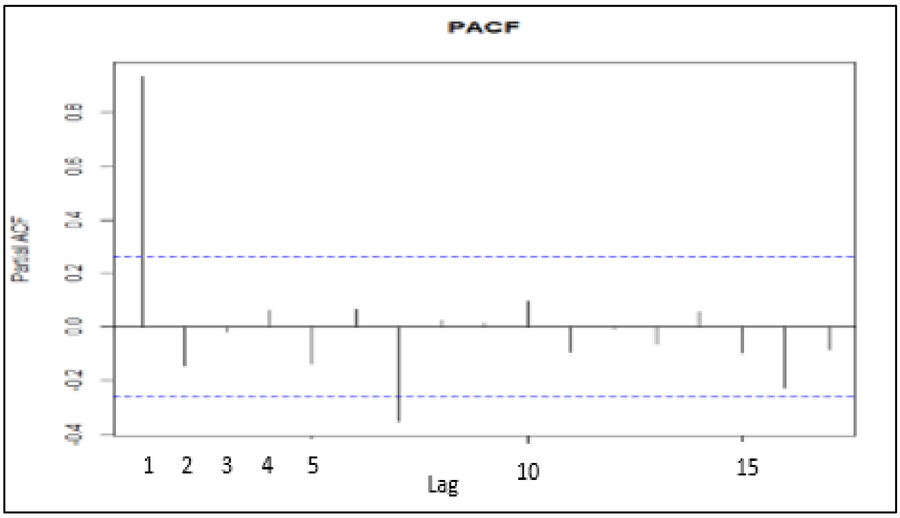

Figure 2. PACF Plot of Composite Stock Price Index (IHSG) Data

Figure 2 shows that the number of *lags* that will be used in this research is 1. This can be seen in the PACF plot which shows a more significant or stronger correlation is only one that is at the 1st *lag*.

3.2 Data Normalization

In the JCI prediction process using the *Sigmoid, ReLU, and Tanh activation functions*, data normalization is carried out. The following are the results of data normalization presented in Table 1.

Table 1. Data Normalization Results

No	Normalized Data
1	0.693
2	0.666
3	0.674
4	0.670
5	0.597
6	0.641
7	0.651
8	0.632
9	0.585
10	0.602
⋮	⋮
56	0.790

3.3 Division of *Training Data* and *Testing Data*

At this stage, the data will be divided into two parts, namely *training data and testing* data, with a percentage of 80% and 20%, where 80% is *training data* and 20% is *testing data*. The results of dividing *training* data and *testing data will* be seen in Table 2.

Table 2. Distribution of *Training Data* and *Testing Data*

Data *Training*	Data *Testing*
0.693	0.861
0.666	0.856
0.674	0.787
0.670	0.784
0.597	0.785
0.641	0.774
0.651	0.807
0.632	0.723
0.585	0.731
⋮	0.812
0.602	0.790

In Table 2 it can be seen that January 2019 – September 2022 is training data with a total of 45 data. Meanwhile, October 2022 – August 2023 is testing data with a total of 11 data.

3.4 Modeling Using the *Extreme Learning Machine* (ELM) Method

The following is the ELM model obtained from *R software* :

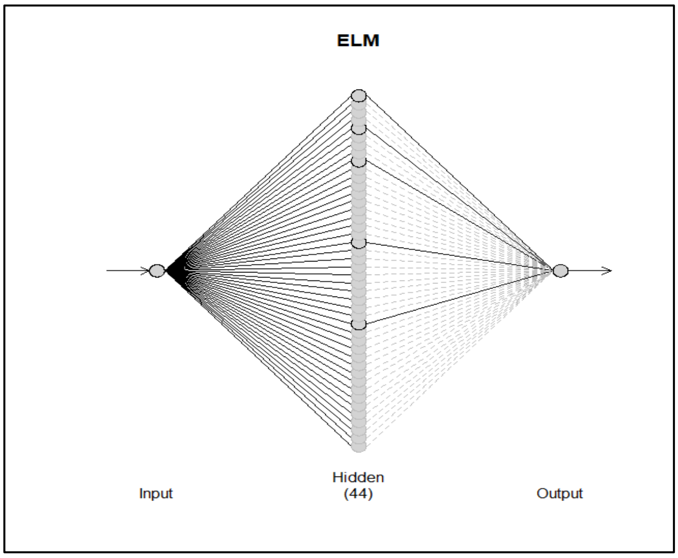

Figure 3. Extreme Learning Machine (ELM) Model

In Figure 4.3 it can be seen that the ELM model has 1 *input neuron*, where 1 *neuron* is IHSG data. There are 44 *hidden layers that appear and the output neurons* that are produced are 1 *neuron* which is the result of the prediction.

- JCI Prediction Using ELM With Activation Function

The activation function used in predicting IHSG uses the *Extreme Learning Machine* (ELM) method, namely *Sigmoid*, *ReLU*, and *Tan h*. The *Extreme Learning Machine (ELM)* method involves several stages of analysis in the process. In the following explanation, we will describe the steps in predicting the composite stock price index (IHSG) using the *extreme learning machine method* with an activation function.

- Calculating IHSG Predictions Using ELM With *Sigmoid Activation Function*

Training Process

The first step taken was to initialize the initial weight values. The initial weight values were generated randomly using *R software* with a value *range* between -1 to 1. The following results of the initial weight initialization can be seen in Table 3.

Table 3. Initialization of Initial Weights

No	W
1	0.044
2	0.9
3	-0.146
4	-0.151
5	0.889
6	-0.096
7	0.665
8	-0.655
⋮	⋮
44	0.369

The next step is to calculate *the hidden layer output* (H_{init}). The following is an example of a manual calculation:

$$H_{init\ 11} = 0{,}693\ x\ 0{,}044 = 0{,}031$$

The following is the result matrix from the *hidden layer output* (H_{init}). For the overall value of the *hidden layer* output matrix (H_{init}):

$$H_{init} = \begin{bmatrix} 0{,}031 & 0{,}062 & -0{,}101 & & 0{,}256 \\ 0{,}030 & 0{,}600 & -0{,}097 & \cdots & 0{,}246 \\ 0{,}030 & 0{,}606 & -0{,}099 & \cdots & 0{,}249 \\ 0{,}030 & 0{,}603 & -0{,}098 & \cdots & 0{,}247 \\ 0{,}026 & 0{,}537 & -0{,}087 & \cdots & 0{,}220 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}039 & 0{,}789 & -0{,}128 & \cdots & 0{,}323 \\ 0{,}036 & 0{,}725 & -0{,}118 & \cdots & 0{,}297 \\ 0{,}036 & 0{,}736 & -0{,}119 & \cdots & 0{,}301 \\ 0{,}039 & 0{,}796 & -0{,}129 & & 0{,}324 \\ 0{,}037 & 0{,}760 & -0{,}123 & & 0{,}311 \end{bmatrix}$$

After finding the *hidden layer output results* ( ), the next step is to calculate the H_{init}*hidden layer* output calculation results using the activation function. One example of a manual calculation is as follows:

$$f(x)_{11} = \frac{1}{1 + e^{-0{,}031}} = 0{,}508$$

The following is the output of *the hidden layer* (H_{init}) using the *sigmoid activation function*. For the overall value of *the hidden layer* output results (H_{init}) use the sigmoid activation function.

$$H_{init} = \begin{bmatrix} 0{,}508 & 0{,}651 & 0{,}475 & & 0{,}564 \\ 0{,}507 & 0{,}646 & 0{,}476 & \cdots & 0{,}561 \\ 0{,}507 & 0{,}647 & 0{,}475 & \cdots & 0{,}562 \\ 0{,}507 & 0{,}646 & 0{,}476 & \cdots & 0{,}561 \\ 0{,}507 & 0{,}631 & 0{,}478 & \cdots & 0{,}555 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}510 & 0{,}688 & 0{,}468 & \cdots & 0{,}580 \\ 0{,}509 & 0{,}674 & 0{,}471 & \cdots & 0{,}574 \\ 0{,}509 & 0{,}676 & 0{,}470 & \cdots & 0{,}575 \\ 0{,}510 & 0{,}690 & 0{,}468 & \cdots & 0{,}581 \\ 0{,}509 & 0{,}681 & 0{,}469 & & 0{,}577 \end{bmatrix}$$

The next step is calculating the output weights from *the hidden layer* to *the output layer*, in this calculation using the *Generalized Inverse Moore-Pensrose Matrix*. The following are the results of the *Moore-Penrose Generalized Inverse matrix* (H^{+}) obtained from *R Software*.

$$H^{+} = \begin{bmatrix} -197{,}104 & -159{,}806 & -171{,}633 & & 12{,}630 \\ -314{,}471 & -253{,}758 & -272{,}963 & \cdots & 13{,}490 \\ 703{,}318 & 570{,}373 & 612{,}535 & \cdots & -45{,}854 \\ 721{,}137 & 584{,}825 & 628{,}055 & \cdots & -47{,}022 \\ -144{,}164 & -115{,}833 & -124{,}775 & \cdots & 3{,}440 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 349{,}036 & 283{,}630 & 304{,}395 & \cdots & -25{,}942 \\ 375{,}280 & 303{,}118 & 325{,}179 & \cdots & -17{,}695 \\ -412{,}025 & -334{,}609 & -359{,}179 & \cdots & 29{,}478 \\ 750{,}870 & 608{,}940 & 653{,}951 & \cdots & -48{,}974 \\ -822{,}436 & -667{,}189 & -716{,}433 & & 54{,}824 \end{bmatrix}$$

Then calculate *the output weight*. The following is an example of calculating *the output weight*.

$$\beta_1 = \big(((-197{,}104)\ x\ 0{,}693) + ((-159{,}806)\ x\ 0{,}666) + ((-171{,}633)\ x\ 0{,}674) + \cdots + (12{,}630\ x\ 0{,}844)\big) = 0{,}326$$

the output weight results :

$$\beta = \begin{bmatrix} 0{,}326 \\ -0{,}051 \\ -1{,}112 \\ -1{,}142 \\ -0{,}195 \\ \vdots \\ -1{,}038 \\ -0{,}010 \\ 1{,}124 \\ -1{,}193 \\ 1{,}568 \end{bmatrix}$$

- Process *Testing*

The first step is to initialize the initial weight values. The initial weight initialization results can be seen in Table 3.

The next step is to calculate *the hidden layer output* (H_{init}). The following is an example of a manual calculation:

$$H_{init\ 11} = 0{,}861\ x\ 0{,}044 = 0{,}038$$

The following is the result matrix from the *hidden layer output* (H_{init}):

$$H_{init} = \begin{bmatrix} 0{,}038 & 0{,}775 & -0{,}126 & & 0{,}318 \\ 0{,}038 & 0{,}770 & -0{,}125 & \cdots & 0{,}316 \\ 0{,}035 & 0{,}709 & -0{,}115 & \vdots & 0{,}290 \\ 0{,}035 & 0{,}706 & -0{,}115 & \cdots & 0{,}289 \\ 0{,}035 & 0{,}707 & -0{,}115 & \cdots & 0{,}290 \\ 0{,}034 & 0{,}697 & -0{,}113 & \vdots & 0{,}285 \\ 0{,}036 & 0{,}726 & -0{,}118 & \cdots & 0{,}298 \\ 0{,}032 & 0{,}651 & -0{,}106 & \vdots & 0{,}267 \\ 0{,}032 & 0{,}658 & -0{,}107 & \cdots & 0{,}270 \\ 0{,}036 & 0{,}730 & -0{,}119 & \cdots & 0{,}299 \\ 0{,}035 & 0{,}711 & -0{,}116 & & 0{,}291 \end{bmatrix}$$

After finding the *hidden layer output results* (H_{init}) in the *testing data, the next step is to calculate the hidden layer* output calculation results using the activation function. One example of a manual calculation is as follows:

$$f(x)_{11} = \frac{1}{1 + e^{-0{,}038}} = 0{,}510$$

The following is the output of *the hidden layer* (H_{init}) using the *sigmoid activation function* :

$$H_{init} = \begin{bmatrix} 0{,}510 & 0{,}685 & 0{,}469 & & 0{,}579 \\ 0{,}509 & 0{,}684 & 0{,}469 & \cdots & 0{,}579 \\ 0{,}509 & 0{,}670 & 0{,}471 & \vdots & 0{,}572 \\ 0{,}509 & 0{,}669 & 0{,}471 & \cdots & 0{,}572 \\ 0{,}509 & 0{,}670 & 0{,}471 & \cdots & 0{,}572 \\ 0{,}509 & 0{,}667 & 0{,}472 & \vdots & 0{,}572 \\ 0{,}509 & 0{,}674 & 0{,}471 & \cdots & 0{,}574 \\ 0{,}508 & 0{,}657 & 0{,}474 & \vdots & 0{,}566 \\ 0{,}508 & 0{,}659 & 0{,}473 & \cdots & 0{,}567 \\ 0{,}509 & 0{,}675 & 0{,}470 & \cdots & 0{,}574 \\ 0{,}509 & 0{,}671 & 0{,}471 & & 0{,}572 \end{bmatrix}$$

The next step is to calculate the predicted output (the output on *the output layer*). An example of a manual calculation of the prediction output is as follows:

$$y_1 = (0{,}510\ x\ 0{,}326) + (0{,}685\ x\ (-0{,}051)) \\ +(0{,}469\ x\ (-1{,}112)) + \cdots + \ = 0{,}861$$

The output results produced at *the output layer* are:

$$y = \begin{bmatrix} 0{,}861 \\ 0{,}856 \\ 0{,}787 \\ 0{,}784 \\ 0{,}785 \\ 0{,}774 \\ 0{,}807 \\ 0{,}723 \\ 0{,}731 \\ 0{,}812 \\ 0{,}790 \end{bmatrix}$$

3.5 Evaluation of Prediction Results

After obtaining *the output layer results* for each activation function (*Sigmoid*, *ReLU*, and *Tanh*), the next step is to obtain the *Root Mean Square Error* (*RMSE) value*. In this calculation, the aim is to evaluate the prediction results to determine the network's ability to recognize a pattern. The results of calculating *the sigmoid* RMSE value can be seen in Table 4, the results of calculating the *ReLU RMSE value* can be seen in Table 5 and the results of calculating the *Tanh RMSE value* can be seen in Table 6.

Table 4. Calculation Results of *Sigmoid RMSE Values*

No	y_i	$\hat{y}_i$	$y_i - \hat{y}_i$	$(y_i - \hat{y}_i)^2$
1	0,8613321	0,8613321	-7 x 10^{-08}	6 x 10^{-15}
2	0,8561038	0,8561039	-1 x 10^{-07}	1 x 10^{-14}
3	0,7874966	0,7874969	-3 x 10^{-07}	1 x 10^{-13}
4	0,7841419	0,7841422	-3 x 10^{-07}	1 x 10^{-13}
5	0,7853017	0,7853021	-3 x 10^{-07}	1 x 10^{-13}
6	0,7740124	0,7740128	-3 x 10^{-07}	9 x 10^{-14}
7	0,8068573	0,8068576	-3 x 10^{-07}	1 x 10^{-13}
8	0,7228537	0,7228539	-2 x 10^{-07}	3 x 10^{-14}
9	0,7313653	0,7313655	-2 x 10^{-07}	4 x 10^{-14}
10	0.8115086	0.8115090	-3 x 10^{-07}	1 x 10^{-13}
11	0.7902594	0.7902597	-3 x 10^{-07}	1 x 10^{-13}
		Amount		8 x 10^{-13}
		$\frac{(y_i - \hat{y}_i)^2}{N}$		7 x 10^{-14}
		$\sqrt{\frac{(y_i - \hat{y}_i)^2}{N}}$		2 x 10^{-07}

Table 5. Calculation Results of RMSE *ReLU Values*

No	y_i	$\hat{y}_i$	$y_i - \hat{y}_i$	$(y_i - \hat{y}_i)^2$
1	0,8613321	0,8613321	1 x 10^{-16}	1 x 10^{-32}
2	0,8561038	0,8561038	0	0
3	0,7874966	0,7874966	2 x 10^{-16}	5 x 10^{-32}
4	0,7841419	0,7841419	1 x 10^{-16}	1 x 10^{-32}
5	0,7853017	0,7853017	1 x 10^{-16}	1 x 10^{-32}
6	0,7740124	0,7740124	0	0
7	0,8068573	0,8068573	0	0
8	0,7228537	0,7228537	1 x 10^{-16}	1 x 10^{-32}
9	0,7313653	0,7313653	1 x 10^{-16}	1 x 10^{-32}
10	0,8115086	0,8115086	3 x 10^{-16}	1 x 10^{-32}
11	0,7902594	0,7902594	1 x 10^{-16}	1 x 10^{-32}
		Amount		2 x 10^{-31}
		$\frac{(y_i - \hat{y}_i)^2}{N}$		2 x 10^{-32}

$\sqrt{\frac{(y_i - \hat{y}_i)^2}{N}}$	1 x 10 $^{-16}$

Table 6. Calculation Results of *Tanh's RMSE Value*

No	y_i	$\hat{y}_i$	$y_i - \hat{y}_i$	$(y_i - \hat{y}_i)^2$
1	0.8613321	0.8613321	- 8 x 10 $^{-9}$	6 x 10 ^{17}
2	0.8561038	0.8561038	- 8 x 10 $^{-9}$	7 x 10^{-17}
3	0.7874966	0.7874966	3 x 10^{-9}	8 x 10^{-18}
4	0.7841419	0.7841419	3 x 10^{-9}	1 x 10^{-17}
5	0.7853017	0.7853017	3 x 10^{-9}	1 x 10^{-17}
6	0.7740124	0.7740124	5 x 10^{-9}	2 x 10^{-17}
7	0.8068573	0.8068573	- 9 x 10^{-9}	8 x 10^{-19}
8	0.7228537	0.7228537	7 x 10^{-9}	5 x 10^{-17}
9	0.7313653	0.7313653	8 x 10^{-9}	5 x 10^{-17}
10	0.8115086	0.8115086	- 2 x 10^{-9}	3 x 10^{-18}
11	0.7902594	0.7902594	2 x 10^{-9}	5 x 10^{-18}
		Jumlah		3 x 10^{-16}
		$\frac{(y_i - \hat{y}_i)^2}{N}$		3 x 10^{-17}
		$\sqrt{\frac{(y_i - \hat{y}_i)^2}{N}}$		5 x 10^{-9}

Based on the results of calculating the RMSE values of the three activation functions above, it can be seen that the activation function *ReLU* produces the lowest RMSE value, namely 1 x 10 $^{-16}$, meaning the *ReLU activation function* is the best activation function in predicting the composite stock price index (IHSG) using the *Extreme Learning Machine method*. The following is a plot of prediction results using the *ReLU activation function.*

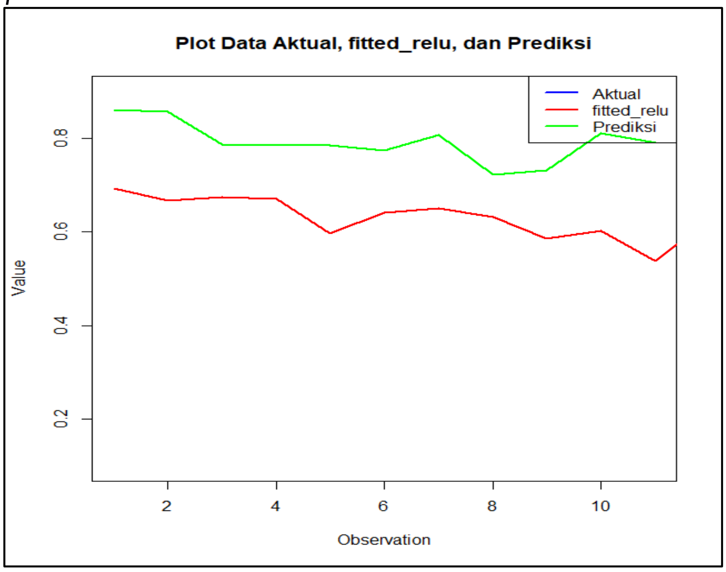

Figure 4. Composite Stock Price Index (IHSG) Data Plot and Prediction Results Using the *ReLU Activation Function*.

Based on Figure 4. 4 It can be seen that the actual data and predicted data overlap each other (following the actual data pattern). This indicates that the model produced from the *ReLU activation function* using the *Extreme Learning Machine* (ELM) method is able to capture actual data patterns very well.

4. CONCLUSION

Based on the results and discussions that have been carried out previously, the best activation function obtained in artificial neural networks using the Extreme Learning Machine (ELM) method to predict the Composite Stock Price Index (IHSG) in Indonesia is the ReLU Activation Function and the evaluation results of the Stock Price Index model The combined (IHSG) for the ReLU activation function produces an RMSE value of 1 x 10-16. These results show that the model's prediction performance in estimating actual values is very good.

5. REFERENCES

Alfiyatin, AN, Mahmudy, WF, Ananda, CF, & Anggodo, YP (2018). Implementation of Extreme Learning Machine (Elm) for Forecasting Inflation Rates in Indonesia Implementation of Extreme Learning Machine for Inflation Forecasting in Indonesia. *Journal of Information Technology and Computer Science (JTIIK)*, *6* (2), 179–186. https://doi.org/10.25126/jtiik.20186900 .

Basit, A. (2020). The Influence of World Gold and Oil Prices on the Composite Stock Price Index (IHSG) for the 2016-2019 Period. *Journal of Accounting Applications*, *5* (1), 42–51. https://doi.org/10.29303/jaa.v5i1.89 .

Devi, S.S. (2021). The Influence of Inflation and Exchange Rates on the Composite Stock Price Index (IHSG) Listed on the Indonesia Stock Exchange (BEI) During the Covid-19 Pandemic in January-December 2020. *JIMEN Innovative Journal of Management Students*, *1* (2), 139– 149.

Fardhani, AA, Simanjuntak, DIN, & Wanto, A. (2018). Prediction of Retail Rice Prices in Traditional Markets in 33 Cities in Indonesia Using the Backpropagation Algorithm. Infomedia Journal, 3(1). https://doi.org/10.30811/jim.v3i1.625.

Fenny, D. (2019). Comparative Analysis of Cosine Normalization and Min-Max Normalization in Grouping Translations of Al-Quran Verses using the K-Means Clustering Algorithm. Journal of Chemical Information and Modeling, 1–115.

Halimi, Imam, and Wahyu Andhyka Kusuma. 2018. "Prediction of the Composite Stock Price Index (IHSG) Using the Neural Network Algorithm." Journal of Informatics Education and Research (JEPIN) 4(1). doi: 10.26418/jp.v4i1.25384.

Divine, S., & Nilogiri, A. (2018). Implementation of Deep Learning in Identifying Plant Types Based on Leaf Images Using a Convolutional Neural Network. JUSTINDO (Indonesian Journal of Information Systems and Technology), 3(2), 49–56.

Karyadiputra, E., & Hijriana, N. (2018). Analysis of the Application of the Naive Bayes Algorithm to Classify Road Development Priorities in South Kalimantan Province. Technologia: A Scientific Journal, 9(2), 105. https://doi.org/10.31602/tji.v9i2.1374.

Mardiana, Ll. (2020) 'Optimization of the Combination of Carbomer and CMC Na in Hair Coloring Gel Preparation of Butterfly Butterfly Extract (Clitoria ternatea L.)'.

Purba, RH, Zarlis, M., & Gunawan, I. (2020). Backpropagation Algorithm in Predicting Poverty Rates in North Sumatra Province. TIN: Applied Informatics Archipelago, 1(1), 55–63.

Roihan, A., Sunarya, PA, & Rafika, AS (2020). Utilization of Machine Learning in Various Fields:

Review paper. IJCIT (Indonesian Journal on Computer and Information Technology), 5(1), 75–82. https://doi.org/10.31294/ijcit.v5i1.7951 .

Sudarsono, A., 2016. Artificial Neural Networks to Predict Population Growth Rates Using Methods. Media Infotama, 12 (1), 6169.

Widodo. (nd). Analysis of the Influence of the Asian Regional Composite Stock Price Index on the Indonesian Composite Stock Price Index. Journal of Economics and Business.