JURNAL MATEMATIKA, STATISTIKA DAN KOMPUTASI e–ISSN: 2614–8811
Published by Departement of Mathematics, Hasanuddin University, Indonesia
https://journal.unhas.ac.id/index.php/jmsk/index
Vol. 21, No. 3, May 2025, pp. 775-785
DOI: 10.20956/j.v21i3.43522

Forecasting the Number of Domestic Flight Passengers at Minangkabau International Airport Using Sliding Window-Based Backpropagation

Leofany Zahra*[1], Tessy Octavia Mukhti[2]

[1,2]Department of Statistics, Padang State University, Indonesia
***Email**: [1]leofanyzhr@gmail.com, [2]tessyoctaviam@fmipa.unp.ac.id*
**Corresponding Author*
Received: 14 March 2025, revised: 19 April 2025, accepted: 20 April 2025

Abstract

This study discusses the forecasting of domestic passenger numbers at Minangkabau International Airport using an Artificial Neural Network (ANN) with a Backpropagation algorithm based on the Sliding Window technique. The data used comes from BPS West Sumatra Province for the 2018-2023 period. The Sliding Window technique transforms time series data into cross-sectional data, which is then modeled using ANN with variations in the number of neurons and hidden layers. The results show that the best model uses 1 hidden layer, 5 hidden neurons, a learning rate of 0.01, and a window size of 5, with an MSE of 0.0027 and a MAPE of 0.0860%. This model has proven to be highly accurate and can be used as a decision-making tool for airport capacity management and operations.

Keywords: Artificial Neural Network, Backpropagation, Minangkabau International Airport, Passenger Forecasting, Sliding Window.

1. INTRODUCTION

Transportation is one of the essential aspects that supports human activities and economic growth [16]. Minangkabau International Airport, as the main gateway to West Sumatra Province, plays a strategic role in facilitating the mobility of people, particularly the increasing number of domestic passengers each year. Along with this growth, managing infrastructure, flight schedules, and capacity planning has become a major challenge for airport operators [1]. Therefore, having accurate passenger forecasts is crucial to support operational planning and enable more effective decision-making.

In several previous studies, Artificial Neural Networks (ANN) have been widely applied for time series forecasting due to their ability to capture complex and nonlinear patterns in data. One relevant study is presented by [15], which successfully employed the Backpropagation algorithm combined with the Sliding Window technique to predict COVID-19 cases, achieving strong performance with the lowest Sum Squared Error (SSE) of 0.69. The Sliding Window technique helps optimize the use of historical data in recognizing trend patterns, thereby enhancing prediction accuracy in time series analysis.

However, although this method has shown promising results in various fields, such as health forecasting and economic data, the implementation of the ANN and Sliding Window combination for predicting air passenger traffic — especially at regional airports like Minangkabau International Airport — is still rarely addressed in the literature. Most previous research has focused on large-scale international airports or used conventional prediction methods such as ARIMA or linear regression, which tend to have limitations in handling the nonlinear nature of passenger flow data. For example, a study by Prawati Ningsih et al. (2019) demonstrated that the SARIMA model was more effective in forecasting the number of international tourist arrivals to West Sumatra through Minangkabau International Airport. In that study, the SARIMA (1, 0, 1)(2, 1, 0)12 model produced predictions that were close to the actual data, indicating its capability to represent seasonal patterns and complex fluctuations in time series data.

Nevertheless, although SARIMA excels in capturing seasonal patterns, this method has certain limitations, particularly in responding to sudden changes in data patterns and its dependence on a complex parameter adjustment process. On the other hand, the Artificial Neural Network (ANN) method using a sliding window approach offers flexibility in handling nonlinear data and adaptability to pattern changes. ANN enables deeper learning from historical data, and as more new data becomes available, the model is able to continuously improve its prediction accuracy (Haykin, 1999).

For this reason, this study aims to apply the Backpropagation algorithm combined with the Sliding Window technique to predict the number of domestic passengers at Minangkabau International Airport. This research is expected to contribute practical insights for airport management in formulating operational strategies, while also enriching the literature by presenting a more adaptive and accurate prediction approach within the context of regional airport passenger forecasting.

2. MATERIAL AND METHODS

A. Types and Sources of Data

This study utilizes secondary data sourced from the official website of Badan Pusat Statistik (BPS) West Sumatera, which can be accessed at the following link: https://sumbar.bps.go.id/id/statistics-table/2/MjUxIzI=/jumlah-penumpang-pesawat-terbang-dalam-negeri-dan-luar-negeri-yang-datang-dan-berangkat-di-bandara-internasional-minangkabau-menurut-bulan--orang-.html

The dataset contains monthly data on the number of domestic and international passengers at Minangkabau International Airport form January 2018 to December 2023. However, this study specifically focuses only on domestic passenger data , as the aim is to predict the number of domestic passengers using artificial neural networks. The features used in this study are:

Table 1. Features

Feature Name	Description
Month	The recorded month of passengers data (January to December)
Year	The year of observation (2018-2023)
Total Domestic Passengers	Total number of domestic passengers (arrivals + departures) recorded each month

B. **Sliding Window Algorithm**

The Sliding Window method is a technique with a fixed-size window that moves around a specific point in a time series to form a defined interval, which is then used for statistical calculations on data of a predetermined length [4]. In this study, the windowing technique is applied to transform time series data into cross-sectional data. Subsequently, the segmented data is modeled using a neural network algorithm.

C. **Artificial Neural Network (ANN)**

Artificial Neural Network (ANN) is an approach in artificial intelligence developed based on the working principles of the human brain. ANN consists of multiple interconnected neurons that can learn from data to predict patterns or perform classification. Each neuron in the network receives signals from other neurons, processes these signals, and transmits the output to the next neuron [12].

1. **ANN Architecture**

Artificial Neural Network (ANN) consists of three main layers:

- **Input Layer**: Receives the data to be processed, where each neuron represents a feature of the data.
- **Hidden Layer**: Consists of one or more layers that process data through interconnected neurons to extract patterns from the input.
- **Output Layer**: Produces the final model output, such as predictions or classifications [12].

The architecture of ANN is illustrated in Figure 1.

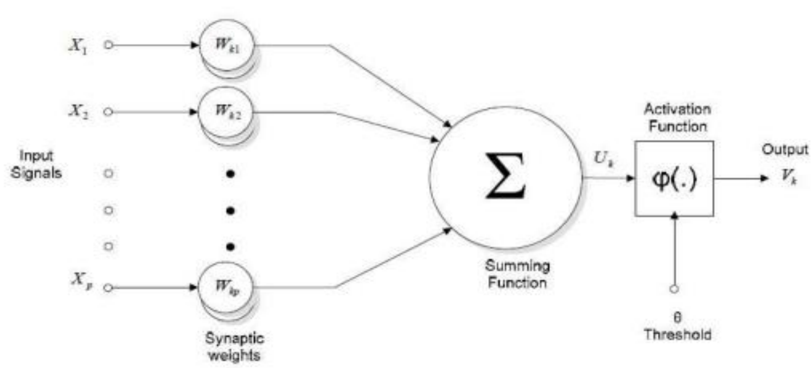

Figure 1. Activation Functions (Source: Fausset, 1994)

2. **Backpropagation Neural Network (BPNN) Algorithm**

Backpropagation Neural Network (BPNN) is one of the most commonly used types of artificial neural networks for modelling and prediction in various applications, such as pattern recognition, classification, and regression. BPNN mimics the learning mechanism of the human brain, where information is processed through interconnected neurons [11]. The training process of BPNN consists of two main phases: forward propagation and backward propagation. In forward propagation, input data is passed through multiple layers of neurons to generate an output. Each neuron computes its output based on the received input, weights, and activation function. In backward propagation, after the output is produced, the error is calculated by comparing the predicted output with the target value. This error is then propagated back through the network to update the weights and biases using optimization algorithms such as gradient descent [5]

The activation function in neural networks determines whether a neuron will be activated or not [8]. Activation functions can be reprocessed and used to represent output values. Common activation functions include sigmoid, tanh, and ReLU (Rectified Linear Unit) [14]. The binary sigmoid function is particularly suitable for the backpropagation algorithm because it has a value range between 0 and 1, making it ideal for networks with outputs within this range. The equation for the binary sigmoid function is as follows.

$$y = f(x) = \frac{1}{(1 + e^{-x})} \tag{2.1}$$

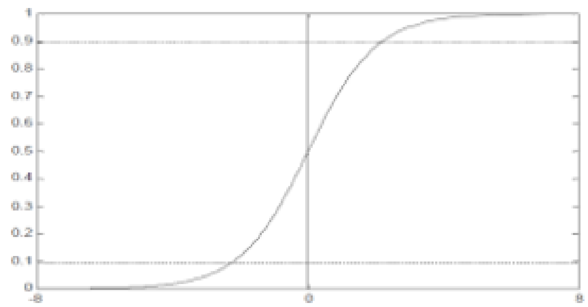

Figure 2. Activation Functions (Source: Fausset, 1994)

The following are the learning stages of the backpropagation algorithm according to Fausset [3].

a. Initialization

- **Step 0** : Initialize the weights randomly within a small range, such as between -1 and 1. Then, set the maximum number of epochs, target error value, and learning rate. Initialize the epoch count to 0 and the Mean Squared Error (MSE) to 1.
- **Step 1** : Proceed to the next steps if the epoch count is still below the maximum and the MSE is greater than the target error. Increase the epoch count by one.
- **Step 2** : Each pair of elements undergoes the learning process before proceeding to the next step.

b. Fase I Feedforward

- **Step 3** : Each input unit (X_a, a = 1, 2, 3, ..., p) receives the input signal X_a and transmits it to all units in the upper layer (hidden layer).
- **Step 4** : Each hidden unit ($Z_{net\ b}$, b = 1, 2, 3, ..., q) accumulates the input signals multiplied by their corresponding weights.

$$Z_{net\ ab} = v_{0b} + (\sum_{ab}^{p} x_a * v_{ab}) \tag{2.2}$$

Notation Explanation:
$Z_{net\ ab}$ = Computed value of the hidden layer
x_{ab} = Input value of the a^{th} input
v_{ab} = Weight between the input and the hidden layer
v_{0b} = Bias weight between the input and the hidden layer
To determine the output signal, the activation function is used:

$$Z_{ab} = f(z_{net\ ab}) = \frac{1}{1 + e^{-Z_{net\ ab}}} \tag{2.3}$$

Notation Explanation:

Z_{ab} = Activation value of the unit $Z_{net\ ab}$
$Z_{net\ ab}$ = Computed value for the hidden layer
The signal is then forwarded to all units in the next layer (output units), and the output values Y_{bc} (where bc = 1, 2, 3, …, p) are calculated.

$$Y_{net\ bc} = w_{0c} + (\sum_{bc=1}^{p} w_{bc} * Z_{ab}) \quad (2.4)$$

Notation Explanation:
$Y_{net\ bc}$ = Input for the output unit Y_{bc}
w_{0c} = Bias weight connection to the output unit Y_{bc}
w_{bc} = Weight connection from the hidden unit Z_{ab} to the output unit Y_{bc}
The output is calculated using the activation function, as shown in Equation 2.4.

$$Y_{bc} = f(Y_{net\ bc}) = \frac{1}{1 + e^{-Y_{net\ bc}}} \quad (2.5)$$

Notation Explanation:
Y_{bc} = Output value
$Y_{net\ bc}$ = Input received by the output unit Y_{bc}

Fase II Backforward
Step 5 : Each output unit (Y_{bc}, bc = 1, 2, 3, … , r) receives the target value corresponding to the input learning pattern and then calculates the error using the following formula:

$$\delta_{bc} = (Y'_{bc} - Y_{bc})f'(Y_{net\ c}) = (Y'_{bc} - Y_{bc})Y_{bc}(1 - Y_{bc}) \quad (2.6)$$

Notation Explanation:
δ_{bc} = Error at the output
Y'_{bc} = Expected target value
Y_{bc} = Output value
The weights are updated by calculating the correction as follows (used to adjust the value of W_{ab}):

$$\Delta w_{bc} = \alpha \delta_c z_{ab} \quad (2.7)$$

Notation Explanation:
Δw_{bc} = Difference between w_{bc} (t) and w_{bc} (t+1)
α = Learning rate
δ_{bc} = Output error
Z_{ab} = Activation value of the unit $Z_{net\ ab}$
The bias correction is calculated using the following formula (used to adjust the value of w_{0c}):

$$\Delta w_{0c} = \alpha \delta_{bc} \quad (2.8)$$

Notation Explanation:
Δw_{0c} = Difference between w_{0c} (t) and w_{0c} (t+1)
α = Learning rate
δ_{bc} = Output error

Step 6 : Each hidden unit Z_{ab} (ab = 1, 2, …, q) accumulates the input received from the upper layer using the following formula:

$$\delta_{net\ bc} = (\sum_{c=1}^{r} \delta_{bc} * w_{bc}) \quad (2.9)$$

Notation Explanation:
$\delta_{net\ ab}$ = Error used to compute the hidden layer error.
w_{bc} = Weight from hidden unit Z_{ab} to output unit Y_{bc}

The error information is computed by multiplying it with the derivative of the activation function:

$$\delta_{ab} = \delta_{net\ bc}\, f\,'(Z_{net\ ab}) = \delta_{net\ bc} \quad (2.10)$$

Notation Explanation:
δ_{ab} = Activation value used to compute the hidden layer error

The weight correction is calculated as follows (used to update the value of v_{ab}):

$$\Delta v_{ab} = \alpha \delta_{ab} x_{ab} \quad (2.11)$$

Notation Explanation:
Δv_{ab} = Difference between v_{ab} (t) and v_{ab} (t+1)
α = Learning rate
δ_{ab} = Hidden layer error
x_{ab} = Input value at unit a
The bias correction is calculated as follows (used to update the value of V_{0b}):

$$\Delta V_{0b} = \alpha \delta_{ab} \quad (2.12)$$

Notation Explanation:
Δv_{0b} = Change in bias between v_{0c} (t) and v_{0c} (t+1)
α = Learning rate
δ_{ab} = Error at hidden layer

Weight and Bias Update
Step 7 : Each output unit (Y_{bc} , bc = 1, 2, …., r) updates its weights and biases (b = 0, 1, 2, …, p) using the equation:

$$w_{bc}(new) = w_{bc}(old) + \Delta w_{bc} \quad (2.13)$$

Similarly, each hidden unit (Z_b, b = 1, 2, 3, …, q) updates its weights and biases (a = 0, 1, 2, …, p) using:

$$v_{ab}(new) = v_{ab}(old) + \Delta v_{ab} \quad (2.14)$$

Step 8 : Calculate MSE (Mean Squared Error)
The obtained weights will be used as the initial weights for the next epoch. This process will continue to iterate until the error reaches the desired target or the maximum number of epochs is reached. The final outcome of the network training consists of the optimized weights, which are stored for model testing and validation.

3. Data Normalization and Denormalization

Normalization is the process of adjusting data to a smaller scale, typically between 0 and 1 [6]. This normalization is necessary to ensure that the data aligns with the range of the sigmoid function. The result of normalization is used to compare the accuracy of predicted values with actual values. Data normalization can be performed using the z-score method, as shown in Equation 2.15 [6].

$$v_a^{`} = \frac{(v_a - A^{-})}{\sigma_A} \quad (2.15)$$

Notation Explanation: $v_a\,'$ = Normalized data v_a = Original data $\bar{A}$ = Mean (average) σ_A = Standard deviation
Denormalization is the process of reverting normalized values back to their original scale to obtain the actual values [2]. Denormalization is performed after the data normalization stage. The denormalization formula is:

$$v_a = (v_a\,' \times \sigma A) + A^{-} \quad (2.16)$$

4. **Model Accuracy**

Mean Square Error (MSE) is the average squared difference between the actual values and the predicted values. MSE is used to measure the error in a predictive model. The smaller the MSE value, the more accurate the prediction results [7]. The formula for calculating MSE is given in Equation 2.17 below:

$$MSE = \frac{\sum_{t=1}^{n}(y_a - t_a)^2}{n} \quad (2.17)$$

Notation Explanation:
n = Number of data points
y_a = Actual value at index a
t_a = Predicted value at index a

Mean Absolute Percentage Error (MAPE) measures the error in the form of an absolute percentage [10]. MAPE is useful for evaluating the accuracy of a predictive model and provides an indication of how large the prediction error is compared to the actual value. The formula for MAPE is given in Equation 2.18.

$$MAPE = \frac{\sum_{t=1}^{n} \left| \frac{(A_t - F_t}{A_t}) \times 100 \right|}{n} \quad (2.18)$$

Notation Explanation:
N = Number of data points
A_t = Actual value at index t
F_t = Predicted value at index t

The smaller the MAPE value, the more accurate the predictive model.

In this study, the ANN model was trained using a backpropagation algorithm. The model development involved trial-and-error to determine the optimal configuration. The architecture was designed with the following parameters:

- Input layer: 5 neurons (determined by the window size).
- Hidden layer: tested using 1 and 2 hidden layers.
- Hidden neurons: 1 to 5 neurons were experimental in each hidden layer.
- Output layer: 1 neuron.
- Learning rate: set at 0,01
- Training epochs: set at 50.000 iterations.

This parameter selection was intended to find the most accurate combination for predicting the number of domestic passengers at Minangkabau International Airport. The final model selection was based on the smallest Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) obtained training and testing.

3. RESULTS AND DISCUSSION

A. Preprocessing

1. **Data Normalization**

The obtained data will be normalized to enable comparison with actual values. Normalization is performed within the interval [0,1] to match the range of the binary sigmoid activation function. An example of the normalized data is presented in Table 1 below.

Table 1. Data Normalization

Input 1	Input 2	Input 3	Input 4	Input 5	Target
0,82000789	0,73607836	0,77179699	0,81182643	0,69658018	1

0,73607836	0,77179699	0,81182643	0,69658018	1	0,94373918
0,77179699	0,81182643	0,69658018	1	0,94373918	0,84140264
0,81182643	0,69658018	1	0,94373918	0,84140264	0,78054007
0,69658018	1	0,94373918	0,84140264	0,78054007	0,77207527
1	0,94373918	0,84140264	0,78054007	0,77207527	0,70556359
0,94373918	0,84140264	0,78054007	0,77207527	0,70556359	0,72425648
0,84140264	0,78054007	0,77207527	0,70556359	0,72425648	0,60390251
0,78054007	0,77207527	0,70556359	0,72425648	0,60390251	0,48815535
0,77207527	0,70556359	0,72425648	0,60390251	0,48815535	0,5334266

2. **Training Process**

During the training phase, the data is divided into two parts: 80% for training and 20% for testing, from a total of 67 data points. A total of 54 data points is used for training with 5 input layers, 1 and 2 hidden layers, 1 output layer, a learning rate of 0.01, and 50,000 epochs.

B. Model Scenario

Several parameters will be tested in this stage, including:

1. The number of neurons in the hidden layer
2. The number of hidden layers.

The experiments are conducted to determine the best artificial neural network architecture for predicting the number of domestic flight passengers at Minangkabau International Airport. The model with the smallest MSE and MAPE values will be selected as the best model. The testing is performed by limiting the number of epochs to 50,000 while varying the number of neurons in the hidden layer, keeping the learning rate fixed at 0.01. A total of 10 different models will be tested using a window size of 5.

C. Testing Results of the Model

The model testing is conducted using 5 inputs based on the predefined window size. The test results include the error percentage of each model, which can be seen in Table 2.

Table 2. Testing Results of the First ANN Model

Input	Hidden Neurons	Output	Hidden Layers	MSE	MAPE (%)
5	1	1	1	0,0030	0,0903
5	2	1	1	0,0035	0,0949
5	3	1	1	0,0028	0,0868
5	4	1	1	0,0029	0,0865
5	5	1	1	0,0027	0,0860

Based on Table 1, it can be observed that the number of neurons in the hidden layer affects the model’s accuracy, as reflected by the variation in MAPE values. The more neurons used, the better the model is able to recognize data patterns, although beyond a certain point, adding too many neurons may lead to overfitting. The combination with the lowest MAPE value is selected as the optimal configuration and is used as a reference for further testing on the model with two hidden layers, as presented in Table 2.

Table 3. Testing Results of the Second ANN Model

Input	Hidden Neurons	Output	Hidden Layers	MSE	MAPE (%)
5	(5,1)	1	2	0,0034	0,0953
5	(5,2)	1	2	0,0034	0,0956
5	(5,3)	1	2	0,0033	0,0909
5	(5,4)	1	2	0,0034	0,0961
5	(5,5)	1	2	0,0035	0,0960

Based on the test results, the model with a single hidden layer and five hidden neurons demonstrated the best performance in predicting the number of domestic passengers at Minangkabau International Airport.

This model achieved a Mean Squared Error (MSE) of 0.0027, indicating that the average squared difference between actual and predicted values is very small. A low MSE value suggests that the model effectively captures patterns in the data and has a negligible error rate. Additionally, the Mean Absolute Percentage Error (MAPE) obtained was 0.0860%, which falls below the 10% threshold, confirming the model's high level of accuracy. In the context of passenger number predictions, a MAPE below 10% indicates that the model is reliable in providing estimates close to the actual values, meaning that the ANN model used can produce predictions with minimal relative error.

When compared to the model with two hidden layers, it was observed that its performance was not superior to the single-hidden-layer model. The MSE and MAPE values for the two-hidden-layer model were slightly higher, suggesting that increasing the number of hidden layers does not necessarily enhance model accuracy. In some cases, the additional complexity introduced by extra layers can lead to overfitting, making the model less optimal when applied to new data.

Based on the prediction results for the number of domestic flight passengers at Minangkabau International Airport using the Backpropagation algorithm with a Sliding Window approach, a projection for the next 48 months has been obtained, as shown in the following table:

Table 4. Predicted Number of Passengers

Month	Number of Passengers
1	178382
2	179399
3	179675
4	179406
5	179612
…	…
48	179151

From the table above, it is evident that the model produces predictions with a stable pattern over the next 48 months. If the number of passengers remains relatively constant, this indicates that the model predicts a stable condition, which could be due to historical trends approaching equilibrium. Conversely, if fluctuations are observed, the model has successfully captured seasonal patterns or recurring trends in historical data.

With an accuracy rate of 99.914%, these results demonstrate that the model performs exceptionally well in mapping historical data patterns and providing projections that can serve as a reference for airport operational planning. However, it is essential to consider external factors such as aviation policies, economic conditions, and changes in travel trends that may affect the actual number of passengers in the future. Therefore, continuous monitoring and periodic model updates are recommended to ensure the prediction accuracy remains optimal.

4. CONCLUSIONS

Based on the findings of this study, it can be concluded that among the tested artificial neural network (ANN) configurations, the model with a learning rate of 0.01, 5 input neurons, 5 hidden neurons, and a single hidden layer produced the most accurate predictions for domestic passengers number at Minangkabau International Airport. The model achieved an MSE of 0.0027 and a MAPE of 0.0860%, indicating a minimal prediction error.

However, it is important to note that this study only evaluated different configurations within the ANN framework and did not include a comparative analysis against other prediction methods such as LSTM, linear regression, or other machine learning algorithms. Therefore, future research is recommended to explore and compare various modeling approaches to confirm the robustness and generalizability of these results.

REFERENCES

[1] Ali, H., Susanto, P. C., & Saputra, F., 2024. Faktor-faktor yang mempengaruhi manajemen transportasi udara: Teknologi informasi, infrastruktur dan kompetensi sumber daya manusia. *Jurnal Sains, Teknologi, dan Logistik (JSTL)*, Vol. 1, No. 4, 154–167. https://doi.org/10.38035/jstl.v1i4.

[2] Fadhilah, R. N., & Ginardi, R. V. H., 2017. Penentuan Harga Dengan Metode Back Propagation pada Aplikasi E-Commerce CariKos Berbasis Web. *Jurnal Teknik ITS*, Vol. 6, No. 2. https://doi.org/10.12962/j23373539.v6i2.23978.

[3] Fausset, L. V., 1994. *Fundamentals of Neural Network: Architecture, Algorithm, and Application*. IEEE Transactions on Computers. https://doi.org/10.1109/TC.1969.222718.

[4] Geng, X., Li, Y., & Sun, Q., 2023. A Novel Short-Term Ship Motion Prediction Algorithm Based on EMD and Adaptive PSO–LSTM with the Sliding Window Approach. *Journal of Marine Science and Engineering*, Vol. 11, 466. https://doi.org/10.3390/jmse11030466.

[5] Gu, W., Guo, B., Zhang, Z., & Lu, H., 2024. Civil Aviation Passenger Traffic Forecasting: Application and Comparative Study of the Seasonal Autoregressive Integrated Moving Average Model and Backpropagation Neural Network. *Sustainability*, Vol. 16, 4110. https://doi.org/10.3390/su16104110.

[6] Han, J., & Kamber, M., 2006. *Data Mining: Concepts and Techniques*, 2nd ed. Burlington, MA: Morgan Kaufmann.

[7] James, G., Witten, D., Hastie, T., & Tibshirani, R., 2012. *An Introduction to Statistical Learning with Applications in R*. Current Medicinal Chemistry, Vol. 7. https://doi.org/10.2174/0929867003374372.

[8] Julpan, Nababan, E. B., & Zarlis, M., 2015. Analisis Fungsi Aktivasi Sigmoid Biner dan Sigmoid Bipolar Dalam Algoritma Backpropagation Pada Prediksi Kemampuan Siswa. *Jurnal Teknovasi*, Vol. 2, 103–116.

[9] Kusumadewi, S., 2004. *Membangun Jaringan Saraf Tiruan (Menggunakan MATLAB & Excel Link)*. Yogyakarta: Graha Ilmu.

[10] Lewis, C. D., 1982. *Industrial and Business Forecasting Methods: A Practical Guide to Exponential Smoothing and Curve Fitting*. London: Butterworth Scientific.

[11] Madhiarasan, M., & Louzazni, M., 2022. Analysis of Artificial Neural Network: Architecture, Types, and Forecasting Applications. *Journal of Electrical and Computer Engineering*, Vol. 2022, Article ID 5416722. https://doi.org/10.1155/2022/5416722.

[12] Mahto, A. K., Alam, M. A., Biswas, R., Ahmed, J., & Alam, S. I., 2021. Short-Term Forecasting of Agriculture Commodities in Context of Indian Market for Sustainable Agriculture by Using

the Artificial Neural Network. *Journal of Food Quality*, Vol. 2021, Article ID 9939906. https://doi.org/10.1155/2021/9939906.

[13] Merdekawati, G. I., & Ismail, 2019. Prediksi Curah Hujan Di Jakarta Berbasis Algoritma Levenberg Marquardt. *Jurnal Ilmiah Informatika Komputer*, Vol. 24, No. 2, 116–128. https://doi.org/10.35760/ik.2019.v24i2.2366.

[14] Mishra, S., Prusty, R., & Hota, P. K., 2021. Analysis of ANN for Time Series Forecasting: Issues and Applications. *International Journal of Forecasting*, Vol. 37, No. 4, 1234–1245.

[15] Radho, A. E., Sugiartawan, P., & Santiago, G. A., 2021. Prediksi Jumlah Kasus COVID-19 Menggunakan Teknik Sliding Windows dengan Metode BPNN. *Jurnal Sistem Informasi dan Komputer Terapan Indonesia (JSIKTI)*, Vol. 4, No. 1, 12–23.

[16] Zulfauzi, R., & Setyawan, Y., 2022. Peramalan Menggunakan Metode Double Exponential Smoothing dan Fuzzy Time Series Cheng (Studi Kasus: Jumlah Penumpang Angkutan Udara Domestik Kota Kendari). *Jurnal Statistika Industri dan Komputasi*, Vol. 7, No. 1, 34–45.