Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 8 No. 4 (2024) 447 - 453 e-ISSN: 2580-0760

Improved Backpropagation Using Genetic Algorithm for Prediction of Anomalies and Data Unavailability

Gunadi Widi Nurcahyo[1*], Akbari Wafridh[2], Yuhandri[3]
[1,2,3]Universitas Putra Indonesia "YPTK" Padang, Indonesia
[1]gunadiwidi@yahoo.com, [2]akbariwfr@gmail.com, [3]yuhandri.yunus@gmail.com

Abstract

Anomalies and data unavailability are significant challenges in conducting surveys, affecting the validity, reliability, and accuracy of analysis results. Various methods address these issues, including the Backpropagation Neural Network (BPNN) for data prediction. However, BPNN can get stuck in local minima, resulting in suboptimal error values. To enhance BPNN's effectiveness, this study integrates Genetic Algorithm (GA) optimization, forming the BPGA method. GA is effective in finding optimal parameter solutions and improving prediction accuracy. This research uses data from the 2022 National Socio-Economic Survey (Susenas) in Solok District to compare the prediction performance of BPNN, Multiple Imputation (MI), and BPGA methods. The comparison involves training the models with a subset of the data and testing their predictions on a separate subset. The BPGA method demonstrates superior accuracy, with the lowest mean squared error (MSE) and highest average accuracy, outperforming both BPNN and MI methods.

Keywords: data prediction; backpropagation; genetic algorithm; survey data

How to Cite: G. Widi Nurcahyo, Akbari Wafridh, and Yuhandri, "Improved Backpropagation Using Genetic Algorithm for Prediction of Anomalies and Data Unavailability", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 8, no. 4, pp. 447 - 453, Aug. 2024.
DOI: https://doi.org/10.29207/resti.v8i4.5507

1. Introduction

Survey data is processed and analyzed to obtain the characteristics of a population. The quality of data produced from survey activities is often assessed by the magnitude of the bias values formed. Bias values can be measured accurately if researchers understand the sources of error that cause bias, namely data unavailability and data anomalies [1]. Data unavailability occurs when there is a lack or loss of data in a survey dataset [2]. The unavailability of data is unavoidable in large-scale surveys [3]. Data anomalies in a survey context refer to inconsistent values that appear in a dataset. Both can cause errors in creating data models, parameter estimates, and forecasting results that are not up to standard [4].

Several methods that can be used to predict anomalies and data unavailability are the MI method [4] and the BPNN method [5]. The MI method can be used to predict data by making a copy of the data set and replacing missing data with estimated values that are close to the true values. The MI method is very good for use in conditions where missing data meets the assumption of missing at random [6]. Missing data occurs in control variables and the value of the output variable is known. Additional variables used as auxiliary variables in predicting missing data must also be available.

The idea of the BPNN method was first put forward in 1961 [7]. The BPNN method works by minimizing the error between the resulting output and the actual target [8]. This process is carried out by updating the network weights and biases based on the error gradient calculated through back-propagation from the output layer to the input layer [9]. The BPNN method is able to provide the best output prediction value with a small average standard error [10].

There are two main weaknesses in calculating the BPNN method, namely low convergence rate and instability [11] This is due to the risk of being trapped in a local minimum condition and the possibility of the minimum error value being very large. A local minimum is a condition where the algorithm will choose the lowest value from a certain function interval and skip the more optimal value from the entire data

Received: 26-10-2023 | Accepted: 08-07-2024 | Published Online: 04-08-2024

function. This method is also computationally time-intensive. Training the BPNN method network can take a long time, especially when using large datasets with many attributes. The process of calculating gradient values and updating weights at each training iteration requires quite large computing resources.

The risk of overfitting is prone to occur in the BPNN method calculations. Overfitting is a condition where the network overly imitates the training data and cannot generalize well to predictions on new data. This risk can occur if the number of hidden layers and neurons is too large [12]. If the number of hidden layers and neurons is too small, the network is at risk of underfitting problems. Both of these things can interfere with the calculation performance of the BPNN method.

To overcome these weaknesses, research was carried out to improve the BPNN method using the GA algorithm. The GA algorithm is a computational method inspired by the theory of evolution and genetics in biology [13]. The GA algorithm operates by manipulating a set of potential solutions through a process of selection, recombination and mutation using probability conditions [14]. The GA algorithm can also be used to make predictions, but the resulting accuracy is not as good as the BPNN method [15].

The GA algorithm plays a role in producing optimal initial weights and biases in calculating the BPNN method [16]. The GA algorithm can also be used in selecting initial parameters for calculating the BPNN method such as the number of hidden layers, momentum value, and learning rate [17]. This improved method is able to produce faster training times and a smaller number of epochs compared to the regular BPNN method [12].

Researchers will analyze the performance of the improved BPNN method using the GA algorithm in predicting anomalies and the unavailability of survey data. Researchers used data from the National Socio-Economic Survey (Susenas) conducted by the Solok Regency Central Statistics Agency (BPS) as research objects. The Susenas survey activity aims to collect data to help formulate public policies and development planning [18].

2. Research Methods

The research began with data preparation. The data from the Susenas results must first be cleaned and normalized. Next, predictions of anomalies and data unavailability are carried out using three methods, namely the BPNN Method, MI Method, and BPNN Method enhanced with the GA algorithm (BPGA). This method comparison was carried out to see the effectiveness of the BPGA method compared to other methods.

2.1 Research Data

This research uses data from the Susenas March 2022 BPS Solok Regency. The data has 600 records originating from a sample of households in Solok Regency. The variable that is the output of the research is the Per Capita Income Variable. The data will first be divided into 2 groups. The first group is training data with a total of 450 records and the second group is testing data with a total of 210 records. The group division was carried out randomly.

The data will go through a cleaning and normalization stage before predictions are made. Data cleaning is carried out to see whether there is empty or duplicate data. Empty and duplicate data must be removed from the data set because it can cause deviations in the analysis results resulting in biased values or wrong conclusions [2].

Data normalization is the process of transforming data into normal form. Data normalization is used to make comparisons between data that have different value domains [19]. This stage can affect the speed, accuracy and learning ability of artificial neural networks. If the data has a different range of values, various problems will arise such as slow data convergence, unstable output results, and local minima [11]. In this research, the output variable will be transformed using the Sigmoid function. The Sigmoid function will change the data into a value range of 0 to 1.

$$f(x) = \frac{0.8 \times (x - Min)}{(Max - Min)} + 0.1 \quad (1)$$

The normalization method that can be used to follow the Sigmoid function is the Adjusted Min-Max method as in Formula 1. The Maximum Value (Max) and Minimum Value (Min) are used as the upper and lower limits of the value [20]. Examples of data conditions before and after normalization can be seen in Table 1. It can be seen that the data is changed according to the value range which follows the Adjusted Min-Max method.

Table 1. Example of comparison of initial data and after normalization

No.	Variabel Code	Initial Data			Data After Normalization		
		Data 1	Data 2	Data 3	Data 1	Data 2	Data 3
1	R105	2	2	2	0,900	0,900	0,900
2	R1804	77	88	64	0,214	0,232	0,193
3	R1805	5	5	5	0,900	0,900	0,900
…	…	…	…	…	…	…	…
37	Output-Capita	1.744.005	1.042.592	956.195	0,232	0,164	0,155

2.2 BPNN Method Prediction

The network structure of the BPNN method consists of multiple layers categorized as input layers, hidden layers, and output layers [11]. This layered architecture allows the neural network to process and learn from complex datasets by passing information through several transformation stages. The structure of the artificial neural network used in this method is illustrated in Figure 1. The input layer is the first layer of the network, where each neuron represents an input feature from the dataset. This layer's primary function

is to receive the raw data and pass it on to the hidden layers for further processing.

The hidden layers are the core processing components of the neural network. They consist of several interconnected processing units known as neurons. Each neuron in a hidden layer receives inputs from the neurons in the preceding layer (which could be the input layer or another hidden layer). The neurons apply weights to these inputs, sum them up, and pass them through an activation function to produce an output. This output is then sent to the neurons in the next layer. The hidden layers enable the network to learn and model intricate patterns in the data by performing non-linear transformations. The depth (number of hidden layers) and the number of neurons in each hidden layer can be adjusted based on the complexity of the problem. The output layer is the final layer of the network. It receives inputs from the last hidden layer and applies weights and an activation function to produce the final output. The number of neurons in the output layer typically corresponds to the number of desired output variables [9].

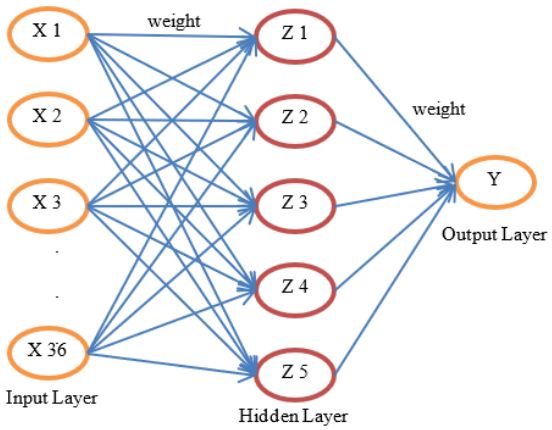

Figure 1. The basic structure of the BPNN Method

The BPNN method comprises four main stages: the Initialized Parameters Stage, the Feed-forward Network Stage, the Backpropagated Error Stage, and the Iteration Stage [7]. In the Initialized Parameters Stage, the initial training parameters are determined. These parameters include the initial weight values, the number of hidden layers, and the bias values. The initial network weights are randomly assigned in the range of 0 to 1 for each input variable.

The Feed-forward Network Stage begins with input data entering the network and propagating through the hidden layers to the output. The hidden layers contain multiple neurons that are interconnected. Each neuron calculates an output value based on the input it receives and the associated connection weights.

After the neuron values in the hidden layers are calculated, an activation function is applied to map the input received by each neuron to the output that will be transmitted to the next neuron. This function introduces a non-linear element to the neural network, enabling it to model complex relationships between inputs and outputs. The choice of activation function depends on the problem's nature and specific preferences. Common activation functions include the Sigmoid/Logistic function, Hyperbolic Tangent/Tanh function, and Rectified Linear Unit (ReLU) function. Each activation function has unique characteristics and impacts on network performance. Therefore, selecting the appropriate activation function is crucial for achieving good results in BPNN method network training.

In the backpropagated error stage, the error value between the real output target and the calculation results of the previous stage is calculated. The iteration stage updates the weights and biases based on error gradients calculated iteratively. The repetition process is carried out until a certain epoch value is reached or when the accuracy target has been achieved. The goal of repetition is to get the smallest difference between the output produced by the network and the actual target value.

Calculations use the R programming language with the Neuralnet library. To see the effect of initial parameter values on network effectiveness, artificial neural network training was carried out four times with different learning rate parameter values, namely 0.01, 0.03, 0.05, and 0.08. The number of hidden layers is determined as 5.

2.3 MI Method Prediction

The MI method uses an iterative process to obtain appropriate prediction data. There are three stages in calculating the MI Method [21]. The first stage is model formation. Build statistical models that can explain the relationship between variables related to missing data and other variables. This model can be a Regression Model, Linear Model, or other model that suits the characteristics of the data. The second stage is repeated imputation. Missing data will be replaced with estimation results generated from the model. This process is carried out several times to produce several complete datasets. The third stage is combined analysis. This stage carries out an analysis of each data set in full. The results of the analysis on each data set are then combined using combining rules to produce final results that take into account the uncertainty of data imputation.

There are several weaknesses that need to be considered in calculating the MI method. First, some predictive variables may be missing from the imputation procedure. This occurs when the prediction variables used to perform the imputation procedure also experience missing data. Second, the data distribution is not normal. The MI method requires the assumption that the data has a normal distribution. Data that is not normally distributed will produce biased values which will greatly influence the results of the analysis. Third, the assumption of the unavailability of MAR data is not met. The MI method can provide inappropriate results due to large bias values. Fourth, computing process

problems. This method requires large computational calculations. Some algorithms require repeated calculations to produce the best value.

This method's calculations use the Mice library in the R language. Different from the previous method, this method requires the output variable in the testing data to be empty. Next, the Mice Function will be run to make predictions on the empty data. To see the influence of parameters, predictions were carried out three times with different numbers of imputation parameters, such as 5, 15, and 30.

2.4 GA Algorithm

GA algorithm applies the principle of survival of the fittest and considers the collection of solutions as a population. Populations will continue to evolve with various forms of genetic operations such as selection, crossover and mutation. Evolution aims to eliminate solutions that have poor fitness values and search for optimal solutions that meet the requirements [9].

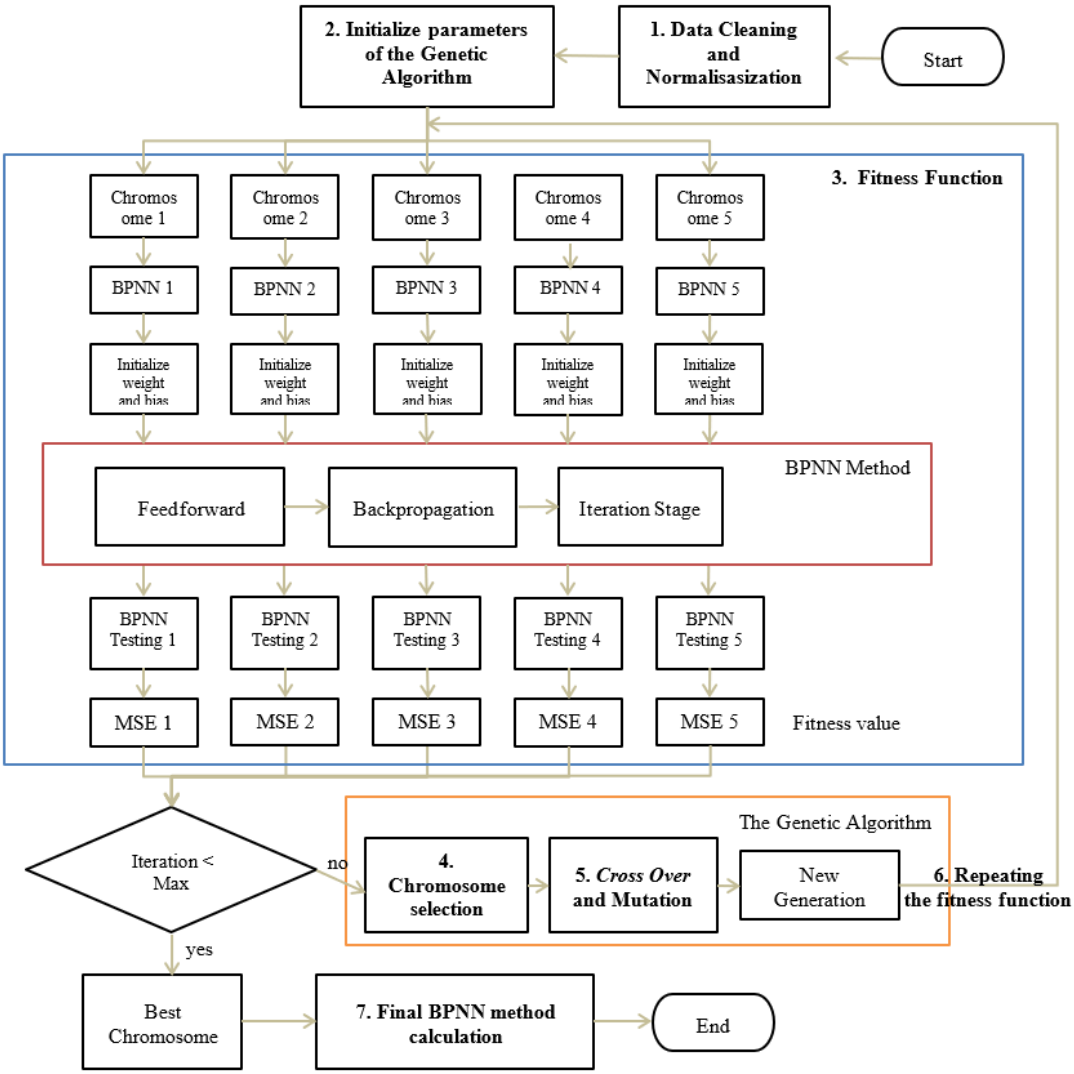

Figure 2. BPGA Method Flow

Genetic Algorithms have strong flexibility and extensive optimization capabilities. This is because this algorithm does not have special limitations and requirements for its use. These advantages make Genetic Algorithms widely used in various fields such as optimization, neural network training, pattern recognition, and time series data prediction.

Genetic Algorithms have proven to be useful in the field of classification. This is due to the ability of this algorithm to explore large and complex search spaces to produce optimal solutions [22]. Genetic Algorithms are also able to avoid being trapped in local optima. Searching for solutions in Genetic Algorithms uses genetic selection and recombination mechanisms that allow variation in the solution population. Various possible solution areas can be covered and potentially better solutions can be found.

2.5 BPGA Method Prediction

The BPGA method combines Genetic Algorithms (GA) and Backpropagation Neural Networks (BPNN) to optimize the selection of input variables for producing

accurate output values. This method uses the best feature selection approach, meaning that not all variables will be involved in constructing the artificial neural network. Instead, the GA algorithm selects the most significant variables, and the BPNN method acts as the fitness function within the GA algorithm [22].

The flow diagram of the BPGA method is illustrated in Figure 2. The process begins with data cleaning and normalization, followed by determining the initial parameters for the GA algorithm, including the number of chromosomes and the dataset size for calculating the fitness function in both the BPNN method and the final BPNN method. Chromosomes, represented as binary numbers, symbolize the use of variables in the neural network architecture.

The subsequent stage involves running the GA algorithm's fitness function, which generates fitness values for each chromosome. Each chromosome has a unique network architecture with a varying number of input variables. The Mean Squared Error (MSE) obtained from testing the BPNN method is calculated for each network architecture and used as the fitness value of each chromosome.

Chromosome selection is based on these fitness values, with selected chromosomes undergoing crossover with other chromosomes to form new ones. Some chromosomes undergo mutations in certain genes, resulting in a new generation of chromosomes. This iterative process repeats until the chromosome with the best fitness value across all generations is identified. The variable arrangement of this optimal chromosome is then used for calculating the final BPNN method, yielding the final prediction results.

This method's calculations utilize the GA library and the Neuralnet library in R. Initially, the fitness function for the GA algorithm is defined, where the MSE from BPNN method predictions serves as the fitness value. The GA algorithm's optimization process identifies the best variables influencing the output, which are then employed in building the final artificial neural network using the BPNN method. In the BPGA method, the BPNN parameters include five hidden layers and a learning rate of 0.03.

3. Results and Discussions

The results of experiments carried out using the three methods are shown in Table 2. The effectiveness of the three methods was tested using the MSE value and average accuracy of the testing data. The BPGA method was proven to have the best value among the three methods, with an MSE value of 0.002439 and an average accuracy of 83.69. The best BPNN method calculation results obtained an MSE value of 0.003351 with an accuracy value of 81.70. The best MI method calculation gets an MSE value of 0.003885 and an accuracy of 80.10.

Table 2. Comparison of Prediction Method Results

Method	Parameter	MSE	Average Accuracy (%)
BPNN	Learning Rate: 0.01	0.003143	80.68
	Learning Rate: 0.03	0.003351	81.70
	Learning Rate: 0.05	0.003529	81.12
	Learning Rate: 0.08	0.006329	76.17
MI	*Imputation:* 5	0.003961	80.10
	Imputation: 15	0.003903	79.79
	Imputation: 30	0.003885	79.63
BPGA	*Learning Rate*: 0.03	0.002439	83.69

Changing the learning rate parameters in the BPNN method does not affect the effectiveness of the artificial neural network. This can be seen from the MSE value which does not always decrease when the learning rate value is lowered. Determining the correct BPNN parameter values needs to be done by trial and error. Apart from that, it is possible to get a better value if the number of hidden layers is also changed. Parameter changes in the MI method were proven to have an effect on the MSE value, although the direction was reversed and the change in the MSE value was not significant. If the number of imputations is increased, the MSE value actually gets worse.

Based on the tests carried out, the BPGA method only requires 20 of the 36 input variables to predict results. The number of variables that must be considered in monitoring anomalous data is greatly reduced. Of course, this becomes much easier. The BPGA method is used to check data anomalies by looking at the difference between the actual output and the predicted results. If the difference in values is too large then it is likely that the data is anomalous data. This is because the predicted results is considered the normal condition of the training data population. By using an accuracy limit of 70%, the testing data that indicated anomalies based on the BPGA method was 32 out of 210 records. The lowest accuracy value is 40.10. Data indicated as an anomaly has an average accuracy of 60.89.

The distribution of predicted output data relative to actual values is illustrated in Figure 3. The X-axis represents the actual per capita values, while the Y-axis represents the predicted values. The precision line in the graph indicates the ideal scenario where predicted values perfectly match actual values (Y = X line). The BPGA method's precision line is closest to this ideal line, indicating superior accuracy compared to the other methods.

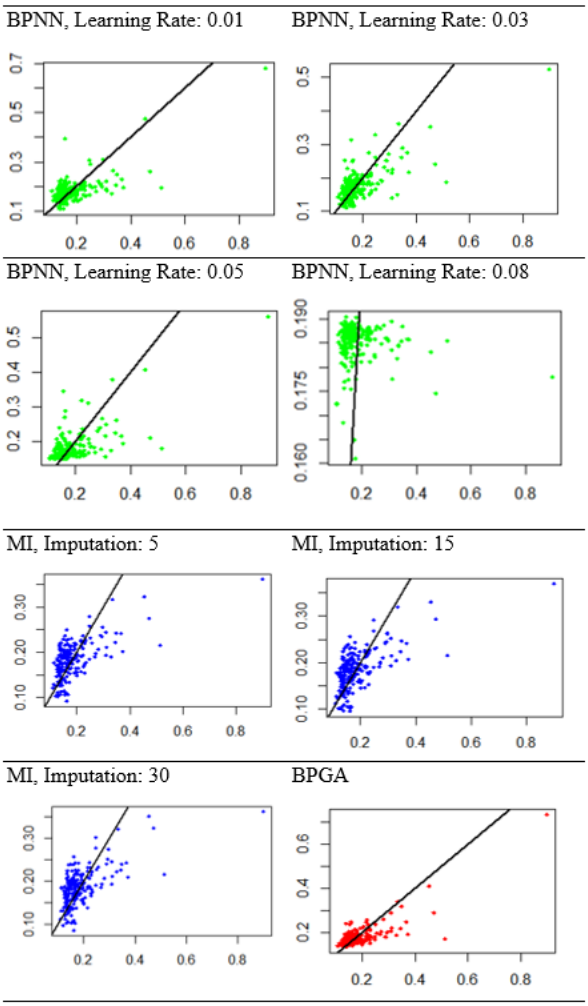

Figure 3. Data Distribution of Method Prediction Results

For actual values within the range of 0 to 0.4, all three methods show relatively centred predictions around the real values. However, for values outside this range, the BPGA method maintains better accuracy with smaller errors. For instance, with an actual output value of 0.9, the BPGA method predicts 0.8, which is significantly closer to the actual value than the BPNN method's prediction of 0.68 and the MI method's prediction of 0.36. This demonstrates that while all methods exhibit some degree of prediction error, the BPGA method consistently provides more accurate predictions, especially for values outside the central range. This indicates the BPGA method's robustness and effectiveness in handling a wider range of data, making it a more reliable choice for predicting anomalies and data unavailability in survey data.

4. Conclusion

Based on research conducted on Solok Regency BPS Susenas 2022 data, the BPGA method has the best predictive effectiveness value. This is indicated by the lowest MSE value obtained at 0.002439 and an average accuracy of 83.69. Researchers can use this method better than the BPNN Method or MI Method in overcoming the problems of anomalies and data unavailability. The performance of the BPNN method can be improved with the GA algorithm by selecting several of the best variables rather than using all variables as input variables. This improved method will be more useful for large data, having many input variables and unknown correlation patterns between variables. This improved method should not be used on little data or not too many input variables. If the correlation pattern between variables is known and it is clear to see its influence on the output variable, it is better to use other methods such as regression.

The data that the author uses in this research is Susenas data which is still raw and still in an unprocessed condition. This research was conducted to assist the data processing stage in searching for anomalous and unavailable data so as to produce clean data. If the goal is to predict data in order to get better accuracy values, it is best that the data used has been processed first.

In future research, it is considered for looking at the influence of other parameters besides learning rate such as the number of hidden layers and maximum epoch in getting more optimal results. Improvements to the BPGA method with approaches other than variable selection could also be considered. Apart from that, there are many other methods that can be combined to improve the BPNN method besides the GA algorithm

References

[1] B. Felderer, A. Kirchner, and F. Kreuter, "The Effect of Survey Mode on Data Quality: Disentangling Nonresponse and Measurement Error Bias," *J. Off. Stat.*, vol. 35, pp. 93–115, Mar. 2019, doi: 10.2478/jos-2019-0005.

[2] P. Kasprzak, L. Mitchell, O. Kravchuk, and A. A. 田文捷 Timmins, "Six Years of Shiny in Research - Collaborative Development of Web Tools in R," *ArXiv*, vol. abs/2101.10948, 2021, [Online]. Available: https://api.semanticscholar.org/CorpusID:231709443

[3] N. Kalpourtzi, J. R. Carpenter, and G. Touloumi, "Handling Missing Values in Surveys With Complex Study Design: A Simulation Study," *J. Surv. Stat. Methodol.*, vol. 12, no. 1, pp. 105–129, Feb. 2024, doi: 10.1093/jssam/smac039.

[4] J. R. Carpenter and M. Smuk, "Missing data: A statistical framework for practice," *Biometrical J.*, vol. 63, no. 5, pp. 915–947, Jun. 2021, doi: https://doi.org/10.1002/bimj.202000196.

[5] A. Kharitonov, A. Nahhas, M. Pohl, and K. Turowski, "Comparative analysis of machine learning models for anomaly detection in manufacturing," *Procedia Comput. Sci.*, vol. 200, pp. 1288–1297, 2022, doi: https://doi.org/10.1016/j.procs.2022.01.330.

[6] G. Vishwakarma, C. Paul, and A. Elsawah, "An algorithm for outlier detection in a time series model using backpropagation neural network," *J. King Saud Univ. - Sci.*, vol. 32, pp. 3328–3336, Dec. 2020, doi: 10.1016/j.jksus.2020.09.018.

[7] C. Sekhar and P. Meghana, "A Study on Backpropagation in Artificial Neural Networks," *Asia-Pacific J. Neural Networks Its Appl.*, vol. 4, pp. 21–28, Aug. 2020, doi: 10.21742/AJNNIA.2020.4.1.03.

[8] Y. Chauvin and D. E. Rumelhart, Eds., *Backpropagation: Theory, architectures, and applications.* in Developments in connectionist theory. Hillsdale, NJ, US: Lawrence Erlbaum Associates, Inc, 1995.

[9] N. Chen, C. Xiong, W. Du, C. Wang, X. Lin, and Z. Chen, "An Improved Genetic Algorithm Coupling a Back-Propagation Neural Network Model (IGA-BPNN) for Water-Level Predictions," *Water*, vol. 11, no. 9. 2019. doi:

10.3390/w11091795.

[10] Y. Lesnussa, C. Mustamu, F. Lembang, and M. Talakua, "Application of Backpropagation Neural Networks In Predicting Rainfall Data In Ambon City," *Int. J. Artif. Intell. Res.*, vol. 2, Aug. 2018, doi: 10.29099/ijair.v2i2.59.

[11] D. Mustikaningrum and R. Wardoyo, "Implementation of Genetic Algorithms and Momentum Backpropagation in Classification of Subtype Cells Acute Myeloid Leukimia," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 14, p. 189, Apr. 2020, doi: 10.22146/ijccs.51086.

[12] J. Tarigan, Nadia, R. Diedan, and Y. Suryana, "Plate Recognition Using Backpropagation Neural Network and Genetic Algorithm," *Procedia Comput. Sci.*, vol. 116, pp. 365–372, 2017, doi: https://doi.org/10.1016/j.procs.2017.10.068.

[13] J. Zhang and S. Qu, "Optimization of Backpropagation Neural Network under the Adaptive Genetic Algorithm," *Complex.*, vol. 2021, pp. 1718234:1-1718234:9, 2021, [Online]. Available: https://api.semanticscholar.org/CorpusID:236150412

[14] Z. Zangenehmadar, O. Moselhi, and S. Golnaraghi, "Optimized planning of repair works for pipelines in water distribution networks using genetic algorithm," *Eng. Reports*, vol. 2, Jun. 2020, doi: 10.1002/eng2.12179.

[15] D. S. R. N. P. R. Suryanita, "Perbandingan Algoritma Genetika dan Backpropagation pada Aplikasi Prediksi Penyakit Autoimun," *Khazanah Inform.*, no. Vol. 5 No. 1 June 2019, pp. 21–27, 2019, [Online]. Available: https://journals.ums.ac.id/index.php/khif/article/view/7173/4605

[16] Y. Hu, A. Sharma, G. Dhiman, and D. M. Shabaz, "The Identification Nanoparticle Sensor Using Back Propagation Neural Network Optimized by Genetic Algorithm," *J. Sensors*, vol. 2021, pp. 1–12, Nov. 2021, doi: 10.1155/2021/7548329.

[17] J. S. Sebayang and B. Yuniarto, "Perbandingan Model Estimasi Artificial Neural Network Optimasi Genetic Algorithm dan Regresi Linier Berganda," *Media Stat. Vol 10, No 1 Media Stat. - 10.14710/medstat.10.1.13-23*, Jun. 2017, [Online]. Available: https://ejournal.undip.ac.id/index.php/media_statistika/article/view/15598

[18] BPS, "Statistik Kesejahteraan Rakyat," Jakarta, 2022.

[19] A. Eesa and W. Arabo, "A Normalization Methods for Backpropagation: A Comparative Study," *Sci. J. Univ. Zakho*, vol. 5, p. 319, Dec. 2017, doi: 10.25271/2017.5.4.381.

[20] I. Purba *et al.*, "Accuracy Level of Backpropagation Algorithm to Predict Livestock Population of Simalungun Regency in Indonesia," *J. Phys. Conf. Ser.*, vol. 1255, p. 12014, Aug. 2019, doi: 10.1088/1742-6596/1255/1/012014.

[21] J. A. C. Sterne *et al.*, "Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls," *BMJ*, vol. 338, 2009, doi: 10.1136/bmj.b2393.

[22] H. Lafta, Z. Hasan, and N. Ayoob, "Classification of medical datasets using back propagation neural network powered by genetic-based features elector," *Int. J. Electr. Comput. Eng.*, vol. 9, p. 1379, Apr. 2019, doi: 10.11591/ijece.v9i2.pp1379-1384.

