Electricity Consumption Modelling in Kendari using The Backpropagation Method on The Artificial Neural Network

R Ruslan¹, L Laome⁴, I Usman² and E W Harisa¹

¹Department of Statistics, Faculty of Mathematics and Natural Sciences, Halu Oleo University, Kendari 93232, Indonesia
²Department of Physics, Faculty of Mathematics and Natural Sciences, Halu Oleo University, Kendari 93232, Indonesia

*E-mail: rushlan_a@yahoo.com

Abstract. Kendari is the capital city of Southeast Sulawesi province with significant population growth. Population growth has resulted in higher demand for electricity, while Kendari city electricity source is only obtained from diesel power plants, of course, the amount of electricity available is limited compared to other power plants. For this reason, it is necessary to model the consumption of electricity in the city of Kendari, as well as forecasting peak loads at certain times of the day. In fact, the assumption of the data cannot be fully fulfilled, even though the assumption has been handled. For this reason, forecasting using the backpropagation method in artificial neural networks provides a solution when the assumptions of statistical data are not fulfilled. The purpose of this article is to model electricity consumption in Kendari city using the back propagation method on artificial neural networks. The best modelling results use a network structure with 10 input layer units and 4 hidden layer units with the smallest mean square error of 0.000145. The highest average peak load on a daily basis occurs at 8.00 PM is 75,593 MWh, while the lowest electricity load that occurs at 05.00 AM is 64,203 MWh. Forecasting of the mean of electricity consumption the next week will produce 69,079 MWh on Monday, 69,381 MWh on Tuesday, 68,550 MWh on Wednesday, 69,124 MWh on Thursday, 68,110 MWh on Friday, 67,927 MWh on Saturday, and 68,833 MWh on Sunday.

1. Introduction

Kendari, the capital city of Southeast Sulawesi province, has a significant population growth. Based on data from the Central Statistics Agency in 2010, the population density per km² was 1067, while in 2018 it had a population growth rate of 3.76%, the population density per km² was 1404 and around 99.23% of electricity sources in Kendari City were very dependent on the availability of electricity at State Electric Company (PLN) [1]. The population growth has resulted in higher demand for electricity, while Kendari city electricity source is only obtained from the diesel power plant from PLN, of course, the amount of electricity available is limited compared to other power plants.

Along with the rapid population growth, of course, it must be balanced with the availability of electricity in the future. Forecasting is the right thing to determine the availability of electricity in the future. Forecasting is done based on historical data in the form of time-series data. The historical data used in this research is electricity consumption data every hour within a period of 30 days, from January to March 2019. Forecasting methods often assume that the data must be stationary and white noise [2]. The assumptions of the data cannot be fully fulfilled, even though the assumptions have been handled.
Research on electricity forecasting or energy consumption has been carried out by previous researchers. Shahriar, Hasan and Al Abrar have discussed a time-series based short-term load forecasting model for the dataset collected from the Regional Power Control Center of a Saudi Electricity Company using an Artificial Neural Network. Artificial Neural Networks perform better than other learning methods [3]. Mandal, Senjyu, Urasaki and Funabashi have developed a practical method for short-term load forecast problems using an artificial neural network (ANN). MAPE values obtained from the load forecasting results confirm that the ANN-based proposed method provides reliable forecasts for several-hour-ahead load forecasting [4]. Jaber, Saleh and Ali have discussed a feed-forward artificial neural network (ANN) that has been selected as an efficient technique to develop a system for accurately predicting building cooling energy requirements per hour for educational buildings at the University of Technology in Iraq [5]. Research on electricity consumption has also been conducted by Gamze, Omer and Selim to discuss forecasting Electricity Consumption with Neural Networks and Support Vector Regression. The forecast results are compared with real consumption values to measure the performance of the methods [6]. Geem and Roper have discussed a neural network model to estimate energy demand in South Korea efficiently [7]. Wang, Gu, Xu and Li have discussed accurate electricity load forecasting algorithms with back-propagation neural networks that are capable of forecasting load efficiently [8].

For this reason, forecasting using the backpropagation method in neural networks provides a solution when the assumption of stationery data and white noise is not fulfilled. The purpose of this article is to model electricity consumption in Kendari city using the backpropagation method on artificial neural networks, as well as predicting peak loads at certain times of the days.

2. Basic Concepts of Neural Networks
The stages of the activation function (\(f(.)\)) used in Artificial neural networks are showing in Figure 1. If a network is to be used for multiple purposes it must have input (will carry the value of an external variable) and output (from a prediction or control signal). Input and output correspond to sensors and motor nerves such as signals coming from the eye and then transmitted to the hand. In this case, there are nerve cells or neurons in the hidden layer that play a role in this network.

The input, hidden layers and outputs of nerve cells are required to connect to each other. Based on the architecture (connection patterns), Artificial Neural Networks can be divided into two categories, namely feed-forward and feed-back (recurrent).

![Figure 1. Activate Function](image)

The feed-forward neural network structure is presented in Figure 2. The feed-forward structure consists of several types, namely [9]:
- a) Single-layer perceptron
- b) Multilayer perceptron
- c) Radial-basic function networks
- d) Higher-order networks
- e) Polynomial learning networks
As with other neural networks, in the feed-forward networks, training is carried out to calculate the weights so that at the end of the training, good weights will be obtained. During the training process, the weights are arranging iteratively to minimize errors that occur. Error (error) is calculated based on the average squared error (MSE). The average of the squares of the error is also the basis for calculating the activation function. Most of the training for the feed-forward network uses the gradient of the activation function to determine how to set weights to minimize performance. This gradient is determined using a technique called Backpropagation [10].

3. Method
The data used in this study is using secondary data obtained from the Kendari branch of the State Electricity Company, namely data from January to March 2019. The variables used in this study are variable \( (Z_t) \) = value of electrical load and variable \( (t) \) = time in the day. The amount of data used is \( (n) = 90 \) days. The steps in this research procedure are [11]:

1. Perform descriptive analysis.
2. Forecasting artificial neural networks using backpropagation, with the following steps:
   a. Perform data transformation using the binary sigmoid function in equation (1).
   b. Divide data into training data and validation test data.
   c. Perform data analysis for the Backpropagation Artificial Neural Network (ANN) model using 3 phases, among others, as follows:
      - Phase 1 is forward propagation, each input signal is propagated (computed forward) form the hidden screen to the output screen using the specified activation function.
      - Phase 2, which is backpropagation, the error (the difference between the network output and the desired target) is propagated backwards from the line directly related to the units on the output screen.
      - Phase 3, namely weight changes, carried out weight modifications to reduce errors that occur.
   d. The forecast results are normalized again so that they can be presented in the form of original or actual data.

4. Result and Discussion
4.1. Descriptive Analysis
Descriptive analysis is an analysis used to assess the characteristics of data. The characteristics in this study are shown in the mean, standard deviation, maximum and minimum values. Following are the results of descriptive statistical analysis of electrical load data can be seen in Table 1.

Based on the table above, it can be seen that the amount of electricity consumption is at least as much as 40.29 MWh. Meanwhile, the largest amount of electricity consumption was 93.78 MWh. The table also shows that the amount of electricity usage for 90 days has increased and decreased every day. This show that the electricity load in Kendari City is uncertain every time. This uncertainty is caused by
various factors such as the increase in population, the rapid development of various industries in Kendari City and other factors. The graph of the peak electricity load in Kendari City can be seen in Figure 3.

Table 1. Descriptive Statistic

|          | Total Electricity Load each day |
|----------|---------------------------------|
| Min.     | 40,29                           |
| Max.     | 93,78                           |
| N        | 90                              |

Figure 3. Average Peak Electricity Load in Kendari City

Based on the graph above, it can be seen that the highest average peak load occurs at 08.00 PM at 82.11 MWh. Meanwhile, the lowest electricity load occurred at 05.00 AM at 54.30 MWh. The time-series data plot can be seen in Figure 4.

Figure 4. Electric Load in Kendari City in January-March 2019

Based on Figure 4, it shows that the highest electricity consumption is 93.78 MWh and the lowest electricity consumption is 40.29 MWh. From the data distribution graph above, the points that are the peak load and decrease of load every day are at relatively the same points. That is, the observations of peak load and decrease load fluctuate around a constant value or point over time. Thus the data from the forecast results in a data pattern similar to that of seasonal data.

4.2. Artificial neural network forecasting with backpropagation
The steps in forecasting Artificial Neural Networks using Backpropagation are as follows:
4.2.1. Data Transformation
The data obtained is transformed first to match the data with the range of the activation function used, namely 0 to 1. The formula used to carry out this transformation is
\[ f'(x) = f(x)(1 - f(x)) \]  
(1)

4.2.2. Data sharing
The data used in this research are electricity consumption data in Kendari every hour for three months from January to March 2019. The data will be divided into 80% of training data and 20% of validation data. So for this research, training data were from January 1 to March 19 and for the target date March 20.

Meanwhile, the data validation is from March 21 to March 30 and the test target is March 31. In this study, the test is to compare the accuracy of a network model that has a maximum MSE of ≤ 0.0001 by projecting data for validation. While validation is matching the projection results against the training data. The projection results that are closer to the original data will be used as the best model for future electrical load projections.

4.2.3. Network Structure Design (Backpropagation Architecture)
The inputted data is 1 to 10 units or more and some hidden display units according to data requirements. Meanwhile, the initialization of the weight or data will occur systematically in the Backpropagation ANN program when it is processed in the Matlab software. The parameter to determine the optimum network structure or not is the fulfillment of the specified MSE targets. To select the smallest MSE and optimum network, training and validation testing of the data will be carried out using the established network architecture.

In this study, the procedure used is specific to the general or bottom-up procedure which is a procedure that starts with a simple model. So that training will begin with one input screen unit and one hidden screen unit. After the first training is complete, a second training is carried out by adding additional units. Training will continue to be carried out by adding units to the input screen and hidden screen until you get the smallest MSE or MSE that matches the target. The process carried out in this training is as follows:

\[ \beta = 0.7(p)^{\frac{1}{n}} \]
\[ \beta = 0.7(1)^{\frac{1}{1}} \]
\[ \beta = 0.7 \]

where \( p \) is the number of hidden units and \( n \) is the number of input units.

a. Initialization of the \( V_{ij} \) weight randomly with the \( V_{ij} \) value is:
\[ -0.5 \leq V_{ij} \leq 0.5 \]
b. Calculating the amount of weight \( V_{ij} \):
\[ \|V_{ij}\| = \sum_{i=1}^{p}(V_{ij})^2 \]
\[ \|V_{ij}\| = \sqrt{\sum_{n=1}^{1}((0,1)^2)} = 0.1 \]
c. Calculating the amount of weight \( V_{ij} \):
\[ v_{ij} = \frac{\beta V_{ij}}{\|V\|} \]
\[ V_{ij} = \frac{0.7 (0,1)}{0.1} = 0.7 \]
d. Set the bias value \( V_{ij} \) to \(-\beta \leq V_{ij} \leq \beta \)
The second step is to provide input and target prices that will become the training data. And for starters starting with one input layer unit, one hidden layer unit and one output layer unit, with the initial input value being 0.50351 which is the data on the first day of January at 00:00 AM.

The next step is to calculate all the network outputs in the output units $Z_j$, $j = 1, 2, ..., m$, using the formula:

$$z_{\text{net}_i} = 0.1 + \sum_{i=1}^{1} (0.50351)0.7 = 0.452457$$

obtained

$$z_i = f(z_{\text{net}_i}) = \frac{1}{1+e^{-0.452457}} = 0.61122324866$$

Then compute all network outputs in output units, $k = 1, 2, ..., m$, with:

$$y_{\text{net}_i} = 0.1 + \sum_{i=1}^{1} (0.61122324866)0.7 = 0.52785627406$$

obtained

$$y_i = f(y_{\text{net}_i}) = \frac{1}{1+e^{-0.52785627406}} = 0.62898298332$$

Next, calculate the $\delta$-unit output factor based on the error in each unit of output, $k = 1, 2, ..., m$, using the formula in the following equation:

$$\delta_k = (t_k - y_k) f'(y_{\text{net}_k})$$

$$\delta_{\text{net}_i} = (0.50351 - 0.62898298332) 0.23336339001 = -0.0292808007$$

obtained

$$f'(y_{\text{net}_i}) = y_i(1-y_i)$$

$$f'(y_{\text{net}_i}) = 0.62898298332 (1-0.62898298332) = 0.23336339001$$

where $t_k$ is target output

$\delta_k$ is unit of error that will be used in deriving the weight of the screen below it.

Calculating the change in weight $w_{kj}$ with rate $\alpha$, using the following equation:

$$\Delta w_{kj} = \alpha \delta_{\text{net}_i} z_j, \quad k = 1,2,\ldots,m; \quad j = 0,1,\ldots,p; \quad \alpha = 0.9$$

$$\Delta w_{kj} = 0.9(-0.0292808007)0.61122324866 = -0.01910739551$$

Calculates the $\delta$ factor hidden unit based on the error in each hidden unit $Z_j$, $j = 1, 2, ..., p$, using the following equation:

$$\delta_{\text{net}_i} = \sum_{k=1}^{m} \delta_k w_{kj} = \sum_{k=1}^{m} (-0.0292808007)(-0.01610739551) = 0.00047163744$$

Calculate the rate of change in weight for $V_i$, using the following formula:

$$\Delta v_{ij} = \alpha \delta_{x_i}, \quad j = 1,2,\ldots,p; \quad i = 1,2,\ldots,n$$

$$\Delta v_{ij} = 0.9(0.00011207492)(0.50351)$$

$$= 0.00005078776$$

obtained

$$\delta_i = \delta_{\text{net}_i} f'(z_{\text{net}_i}) = \delta_{\text{net}_i} z_i (1-z_i)$$

$$= 0.00047163744(0.61122324866)(1-0.61122324866)$$

$$= 0.00011207492$$

Calculates the weight changes. Change inline weight leading to the output unit, namely:

$$w_{ij} \text{ (new)} = w_{ij} \text{ (old)} + \Delta w_{ij}, \quad k = 1,2,\ldots;m; \quad j = 0,1,\ldots,p$$
= (-0.01610739551) + 0.00005078776
= -0.0160566078

- Change inline weight leading to hidden units, namely:
  \[ v_{ij}^{(new)} = w_{ij}^{(old)} + \Delta v_{ij}, \quad j=0,1,2,...; \quad i=0,1,...,n \]
  \[ = (-0.01610739551) + 0.00011207492 \]
  \[ = -0.0159953206 \]

- Calculating the error with the equation:
  \[ E = t_1 - y_1 \]
  \[ E = 0,50351 - 0,62898298332 = -0,1254729833833 \]
  \[ MSE = \frac{\sum(-0,1254729833)^2}{1} = 0,01574346954 \]

For \( n \) further is done in the same way. The above steps are repeating during the training under the conditions of the stop being met. In the training calculation process above, that the MSE obtained is 0.01574346954 and it still does not meet the target, then the next experiment will be carried out. For further experiments with two input layers and two hidden layers and one unit output layer, it was carried out using Matlab software and the following training results are in Figure 5.

![Figure 5. Training 2 Input Layer Units and 2 Hidden Layer Units](image)

In Figure 5, that training using two input screen units and two hidden screen units and 12 validation repetitions has MSE 0.00156 with 12 epochs or is still far from the expected target so that additional input screens and hidden screens will be added as in Figure 6.

![Figure 6. Training 3 Input Layer Units and 2 Hidden Layer Units](image)

In Figure 6, it can also be seen that the mean square error (MSE) of training with three input layer units and two layers units is also still far from the expected target. This MSE training is 0.00119 and 20 epoch so that the input screen unit and hidden screen unit will be added. The following training will conduct experiments on five layers units of input and two units of hidden screens as shown in Figure 7.
In the training, five input layer units and two hidden layer units still have MSE values that are far from the specified target. Therefore, the researcher returned to the training once again to ascertain whether there would be a smaller change in MSE. The MSE of this training is 0.003 and 14 epoch. In these conditions, further training is carried out with additional units. The following training with ten input layer units and four hidden layer units. The results of the training network structure can be seen in Figure 8.

In Figure 8, it can show that MSE training ten input screen units and four hidden screen units (0.000145) and 10 epoch, which means it has MSE that can be used for implementation. Thus based on the picture above, the network architecture chosen is with ten input layer units and four hidden layer units. Furthermore, to see the results of the optimum network selection training can be seen in Table 2. Based on Table 2, it shows that training on 1 to 10 input screens with 1 to 4 hidden screen units has a different MSE value on average, this means that the more input screen units and hidden screen units used, the more accurate the program work results, but that is not always the case because the network structure also has a threshold.

| No. | Input Layer | Hidden Layer | MSE   | No. | Input Layer | Hidden Layer | MSE   |
|-----|-------------|--------------|-------|-----|-------------|--------------|-------|
| 1   | 1           | 1            | 0.0157| 6   | 6           | 2            | 0.00179|
| 2   | 2           | 2            | 0.00156| 7   | 7           | 3            | 0.00111|
| 3   | 3           | 2            | 0.00119| 8   | 8           | 3            | 0.00433|
From the training results (Table 2), it can be that the smallest MSE that meets the target is 0.000145 from the results of training ten input layers and four hidden layers, then the optimum network structure to be used in predicting electrical load is the smallest data, but the first we will testing the accuracy of the network structure by conducting testing experiments on data that has been prepared for validation testing, namely data from March 21 to March 31, 2019. The results obtained on various tests will be compared with the original data and choose the best model or model that has the closest results to the data original for use in forecasting. The following is the validation result data in Figure 9.

After selecting or determining the optimum network structure with the smallest MSE, the next step is to predict the electrical load of Kendari city for the next 3 months (90 days) using the structure or model that has been obtained, namely the ten input layer unit model, four hidden layer units. In this forecasting process, the data used are transformed data from the electric load data obtained. So that the resulting forecast data is none other than data in the form of transformations as well. But after all the forecasting process (ANN training) is complete and the results are out, then the forecast data is normalized again so that it can be presented in the form of original or actual data. Namely by using the formula for linear transformation, namely 

\[
x' = \frac{0.8(x-a)}{b-a} + 0.1
\]

Then to find the data value \(x\), that is

\[
x = \frac{(x'-0.1)(b-a)}{0.8} + a
\]

After the actual data is obtained, then the data is calculated on average and divided into seven days, making it easier to make the graph and make it easier to understand. The following graph of the average electricity load in 1 week (7 days) can be seen in Figure 10.
From the results of forecasting that have been done so that the best ANN model is the model 10 input layers and 4 hidden layer units with the smallest MSE value of 0.000145. So based on Figure (4.8) the highest average peak load daily occurs at 8.00 PM is 75,593 MWh, while the lowest electricity load occurs at 05.00 AM is 64,203 MWh. Forecasting of the mean of electricity consumption the next week will produce 69,079 MWh on Monday, 69,381 MWh on Tuesday, 68,550 MWh on Wednesday, 69,124 MWh on Thursday, 68,110 MWh on Friday, 67,927 MWh on Saturday, and 68,833 MWh on Sunday.

5. Conclusion
Based on the research results, it is found that the best Artificial Neural Network model is the model 10 input layers and 4 hidden layer units with the smallest MSE value of 0.000145. The highest average peak load daily occurs at 8.00 PM is 75,593 MWh, while the lowest electricity load that occurs at 05.00 AM is 64,203 MWh. Forecasting of the mean of electricity consumption the next week will produce 69,079 MWh on Monday, 69,381 MWh on Tuesday, 68,550 MWh on Wednesday, 69,124 MWh on Thursday, 68,110 MWh on Friday, 67,927 MWh on Saturday, and 68,833 MWh on Sunday.

References
[1] BPS 2018 Sulawesi Tenggara Province in Figures 2018 (Kendari: BPS-Statistics of Sulawesi Tenggara Province)
[2] Makridakis S, Wheelwright SC and McGee VE 1983 Forecasting 2nd Edition (New York: Wiley & Sons,Inc)
[3] Shahriar S M, Hasan M K and Al Abrar S R 2019, An effective artificial neural network based power load prediction algorithm Int J Comp Appl 178 20
[4] Mandal P, Senjyu T, Urasaki N and Funabashi T 2006. A neural network based several-hour-ahead electric load forecasting using similar days approach Int J Elect Power 28 6 367–373.
[5] Jaber A A, Saleh A A M and Ali H F M 2019 Prediction of hourly cooling energy consumption of educational buildings using artificial neural network, Int J on Adv Sci Eng Inf Tech 8 1 159-166
[6] Gamze O, Omer F D and Selim 2012, Forecasting electricity consumption with neural networks and support vector regression Proc Soc Behav Sci 58 1576 – 85
[7] Geem ZW and Roper WE 2009, Energy demand estimation of South Korea using artificial neural network, En Policy 37 4049-54.
[8] Wang Y, Gu D, Xu J and Li J 2009 Back propagation neural network for short-term electricity load forecasting with weather features. Int Conf on Comp Intel Nat Comp 1 IEEE 58–61
[9] Siang J J 2005 Jaringan Syaraf Tiruan dan Penrogramannya Menggunakan Matlab (Yogyakarta: Andi-Yogyakarta)
[10] Haykin S 1999 Neural Networks: A Comprehensive Foundation Second Edition (Oxford: Prentice-Hall)
[11] Zhang G P, Patuwo B E and Hu M Y 1988 Forecasting with artificial neural network: The state art Int J Forecast 14 35-62