Application of artificial neural network with backpropagation algorithm for estimating leaf area

E Asriani1* and Robika2

1 Department of Agrotechnology, Faculty of Agriculture, Fisheries, and Biology, Universitas Bangka Belitung, Indonesia
2 Department of Biology, Faculty of Engineering, Universitas Bangka Belitung, Indonesia

E-mail: euis_asriani@yahoo.com

Abstract. Leaf area plays an important role in plant growth and production. There are some methods used to measure leaf area with different approaches such as regression equations, grid count, gravimetric, planimeter, image processing, adaptive neural-based fuzzy inference system, and others. In this research, the excellencies of Artificial Neural Network (ANN) with backpropagation algorithm will be inspected to estimate leaf area of seven plants species. Two parameter, leaf length and leaf width, used as input variables and leaf area as output variable. The result indicated that ANN with 2-50-1 architecture has a good performance in predicting leaf area of seven plants species at 99.99% degree of accuracy.

1. Introduction
Among the most important organs of plants are leaves. Leaves are also a good indicator of plant growth [1]. The primary function of the leaf is to absorb and process sunlight and carbon dioxide for photosynthesis [2]. Solar energy is captured by the leaves and used to drive photosynthetic chemical reactions. The product of this chemical reaction is assimilate which has an important role in plant metabolism [3]. Leaf area growth determines light interception and is an important parameter in determining plant productivity [4].

Leaf area is one of the most important parameters in the analysis of plant growth. Leaf area values are used in the formula for calculating Leaf Area Ratio (LAR), Specific Leaf Area (SLA) and Leaf Area Index (LAI). The leaf area of each plant is highly dependent on the leaf morphology. Leaf morphology has wide variations because leaves have very high plasticity compared to other plant organs [5]. The characteristics of leaves that greatly affect the leaf area is the shape of the leaf. The leaf shape is on of them influenced by the incision found on the leaf edge. Leaf edge can be divided into two main groups, namely the flat edge leaf and the incised edge leaf. The two main groups has a very different leaf shape. Therefore, in calculating leaf area, it is necessary to determine the right method in order to get an accurate leaf area value [6].

There are several methods of measuring leaf area, including measurement methods using millimeter block paper, gravimetric methods, planimeter methods, and photography methods [7]. In addition to these methods, a number of methods with a mathematical approach have been developed including the ANFIS (Adaptive Neural Based Fuzzy Inference System) method [8], and LxW product [9]. There are several advantages of the mathematical method including alternatives which also provide accuracy and speed of measurement compared to traditional methods which are generally tiring and require a long...
time. The ANFIS method is basically an integration between two mathematical approaches namely artificial neural networks and fuzzy logic. The neural network has been applied widely in recent years, including back propagation (BP) neural networks. The use of artificial neural networks in predicting the value of a variable in agriculture has been done in previous studies. Some of these studies include the use of artificial neural networks for prediction of the cotton crop leaf area, prediction of evapotranspiration, leaf area index, and rice productivity [10], [11], and [12]. In this research, a modeling approach is applied that applies one of the two mathematical approaches, namely artificial neural network modeling with BP algorithm. This method is expected to be able to provide a prediction model in determining leaf area with an accuracy level that is at least almost the same as ANFIS method.

2. Research Method

2.1 Leaf Sample Collection

Leaf samples in this study were plant leaves at ultisol land in Bangka Regency which have short and longitudinal irregular morphology. The plants sampled consisted of 7 (seven) plant species, namely palm leaves, maize, thatch, chili, pepper, betel, and kale. From each plant 30 leaf samples were taken, so there were a total of 210 leaf samples. Every 30 leaf samples from each plant are chosen so that they represent the diversity of leaf sizes. For each leaf sample, 3 variables were measured, namely length, width, and leaf area. Length and width are measured using a ruler while the area uses Leaf Area Meter (LAM). These three variables are used to construct artificial neural networks.

2.2 Modeling Leaf Area ANN

The working process of a BP Neural Network can be described by some steps. Define the number of input layer neurons as n, the number of hidden layer neuron H, and the number of output layer neuron m. The connecting weights between the input layer and the hidden layer are $V = (v_1, v_2, \ldots, v_j v_H)$ with column vector $v_j$ standing for the weight vector corresponding to the number j neuron in the hidden layer, $b_j$ standing for the threshold value. The connecting weights between the hidden layer and the output layer are $W = (w_1, w_2, \ldots, w_k, \ldots, w_m)$, with the column vector $w_k$ standing for the weight vector corresponding to the number k neuron in output layer, $b_k$ standing for the threshold value. The input vector is $X = (x_1, x_2, \ldots, x_i, \ldots, x_m)^T$, the output vector of hidden layer $Y = (y_1, y_2, \ldots, y_j, \ldots, y_k)$, the output vector of output layer $O = (o_1, o_2, \ldots, o_k, \ldots, o_m)$, from which we can see that the input data is processed by connecting weights and functions from the input layer, and output data is finally achieved in the output layer; this means that the neural network is a nonlinear mapping from input to output.

According to [13] the details process of BP Neural Network is as follows.

1) Forward transfer process of information. The input data are transferred layer by layer from the input level to the output level, and the data reaching the output layer will be compared with the expected data, and if it does not reach the required error, the error will be transferred backward. The forward transfer process of data is as follows:

For the hidden layer, there is:

$$net_j = \sum_{i=1}^{n} v_{ij} x_i + b_j \quad (j = 1, 2, \ldots, m)$$  \hfill (1)

$$y_j = f(ne_{j}) \quad (j = 1, 2, \ldots, m)$$  \hfill (2)

The input value of input layer is

$$net_k = \sum_{j=1}^{m} w_{jk} c_j + b_{zm}$$  \hfill (3)
The output value is

\[ o_k = f(net_k) \]  

(4)

2) The backward transfer process of errors. The error signal is transferred layer by layer from the output layer to the input layer, during which the error is distributed to neurons of each layer, continuously, producing the error signal of each-layer neurons, and the error signal can correct the weights and threshold values, whose process is as follows.

The error between the expected output value and the real output value of the number \( k \) sample is expressed like this:

\[ E = \frac{1}{2} \sum_{k=1}^{l} (d_k - o_k)^2 \]  

(5)

Unfold the error to the hidden layer and the output layer, during which the error is:

\[ E = \frac{1}{2} \sum_{k=1}^{l} \left( d_k - f \left( \sum_{j=1}^{m} w_{jk} f \left( \sum_{i=1}^{n} v_{ij} x_i \right) \right) \right)^2 \]  

(6)

In order to reduce the network error, the weights and threshold values are adjusted during the training process, and, at this moment, the adjustment of weights is proportional to the gradient descending of error, whose expression is:

\[ \Delta w_{jk} = -\alpha \frac{\delta E}{\delta w_{jk}} \quad (j = 1,2,\ldots,m; k = 1,2,\ldots,l) \]  

(7)

\[ \Delta v_{ij} = -\alpha \frac{\delta E}{\delta v_{ij}} \quad (i = 1,2,\ldots,n; j = 1,2,\ldots,m) \]  

(8)

Wherein, \( \alpha \) is the learning rate, with minus standing for gradient descending. So, this algorithm is also called gradient descending algorithm and, thus, error \( E \) is connected with weights and threshold values that are adjusted through forward and backward transfer, until the weights meet the requirement.

In this study, the design of neural network architecture consists of an input layer, a hidden layer and an output layer. Previous literature reported that one hidden layer was normally adequate to provide a good estimation and can be the first trial for any practical feed forward network design. Therefore, this study used single hidden layer in generating model prediction. The input layer consists of two variables, which are length and width leaf, while the output layer is a leaf area variable. The design of artificial neural networks is shown in Figure 1.

![Figure 1. Backpropagation Artificial Neural Network Design](image)

A number of trial and error was employed to gain the best network performance. Each of them combining some parameters those are number of epoch, goal, and maximum fail with sigmoid as a transfer function. Mean Square Error (MSE) was used to measure function performance.
3. Result and Discussion
Statistics of leaf length, leaf width, and leaf area from seven plants species can be seen in the table 1. The leaf length parameters are at values 6.7 to 86.1, leaf widths from 1 to 8.9 and leaf area from 11.32 to 416.44. The average of the three parameters are 26.6, 3.95, and 75.59 respectively.

| Statistical variable | Independent variables | Dependent variable |
|----------------------|-----------------------|--------------------|
|                      | Leaf length (cm)      | Leaf width (cm)    | Actual leaf area (cm²) |
| Minimum              | 6.7                   | 1                  | 11.32                |
| Maximum              | 86.1                  | 8.9                | 416.44               |
| Average              | 26.6                  | 3.95               | 75.59                |

There is a certain step should be done before employing ANN to train a network, namely normalization the data. The data of this study was normalized using formula:

\[ X' = \frac{0.8(X-x_{\text{min}})}{x_{\text{max}}-x_{\text{min}}} + 0.1 \]  

After normalizing the data, the next step was conducted training on the network based on two input parameters namely the length and width of the leaf and the output parameter was the area of the leaf. By conducting some trial on a number of parameters it was found that architecture 2-50-1 (Figure 2) showed the best model performance among other trials. This architecture means that there are two input parameter (leaf length and leaf width), fifty number of neuron, and one output parameter (leaf area). This architecture produces the best predictions on combinations number of epoch 200, goal 0.000001, and maximum fail 200 with sigmoid as a transfer function.

Based on Figure 3, the solid blue line represents the best fit of linear regression line between output/predicted leaf area and target/measured leaf area. It can be seen that the training data indicates a good fit. The dashed line shows the measured leaf area. The two lines almost coincide with \( R = 0.99288 \). This is an indication of the very high linearity relationship between the predicted and measured leaf.
area, even near to 1. Figure 4 shows the iteration at which the best validation performance reached a minimum. This plot has mean square error $MSE = 0.00014971$.

![Training: R=0.99288](image1)

**Figure 3.** The regression plot of training data between the predicted and measured leaf area

![Best Validation Performance is 0.00014971](image2)

**Figure 4.** The performance progress plot

Comparison between the predicted and measured leaf area of the ANN model is shown in the figure 5. It can be seen that the predicted leaf area has almost the same pattern with the measured leaf area.

![Comparison between predicted and measured leaf area](image3)

**Figure 5.** The comparison between predicted and measured leaf area.

To calculate the accuracy of the constructed ANN model, used Mean Square Error (MSE) as done by [14].

$$\text{accuracy} = (100 - MSE)\%$$  \hspace{1cm} (10)

This ANN model with BP algorithm reach the accuracy at 99.99%.

**Conclusion**

The ANN model with BP algorithm, 2-50-1 architecture at 200 number of epoch, 0.000001 goal, and 200 maximum fail gives a very good prediction accuracy at 99.99%.

**References**

[1] Schongart R, Rocha de M, and De Queiroz H 2011 *Amaz. Floodplain For* pp 465-483

[2] Smith W K, Vogelmann T C, DeLucia E H, Bell D T, and Shepherd K A 1997 *Bioscience* 47 pp
Acknowledgement
We acknowledge the support from University of Bangka Belitung (UBB) for the chance to finish this study.