Analysis of Artificial Neural Network in Erosion Modeling: A Case Study of Serang Watershed

N Arif 1,2*, P Danoedoro 3, Hartono 3
1Doctoral Program in Faculty of Geography, Universitas Gadjah Mada, Sekip Utara Bulaksumur, Yogyakarta, 55281, Indonesia
2Faculty of Science and Technology, Universitas Muhammadiyah Gorontalo, Pentadio Timur, Gorontalo, 96181, Indonesia
3Faculty of Geography, Universitas Gadjah Mada, Sekip Utara Bulaksumur, Yogyakarta, 55281, Indonesia

nursida.arif@um-gorontalo.ac.id

Abstract. Erosion modeling is an important measuring tool for both land users and decision makers to evaluate land cultivation and thus it is necessary to have a model to represent the actual reality. Erosion models are a complex model because of uncertainty data with different sources and processing procedures. Artificial neural networks can be relied on for complex and non-linear data processing such as erosion data. The main difficulty in artificial neural network training is the determination of the value of each network input parameters, i.e. hidden layer, momentum, learning rate, momentum, and RMS. This study tested the capability of artificial neural network application in the prediction of erosion risk with some input parameters through multiple simulations to get good classification results. The model was implemented in Serang Watershed, Kulonprogo, Yogyakarta which is one of the critical potential watersheds in Indonesia. The simulation results showed the number of iterations that gave a significant effect on the accuracy compared to other parameters. A small number of iterations can produce good accuracy if the combination of other parameters was right. In this case, one hidden layer was sufficient to produce good accuracy. The highest training accuracy achieved in this study was 99.32%, occurred in ANN 14 simulation with combination of network input parameters of 1 HL; LR 0.01; M 0.5; RMS 0.0001, and the number of iterations of 15000. The ANN training accuracy was not influenced by the number of channels, namely input dataset (erosion factors) as well as data dimensions, rather it was determined by changes in network parameters.

Keywords: neural network, erosion, watershed

1. Introduction
Soil erosion is a process of soil displacement from one place to another, which becomes an indicator of declining quality of land, thus affecting the sustainability of agricultural production globally [1-4]. Prediction of erosion-prone area is important for evaluation of land use and to support decisions in making regulations on land management. The problem is there is high uncertainty of spatial data,
greatly affecting the accuracy of prediction result, including erosion prediction. Spatial data is very complex and contains a lot of uncertainty, as evident in the nature, collection, and analysis technique of the data.

Problems in the classification of non-linear spatial data with dimension of big data can be solved by artificial neural network approach [5]. The artificial neural network is one machine learning that is widely used in solving various problems of classification and prediction. An artificial neural network is a process of training input data to identify an output pattern, i.e. the target that has been determined by its ability to learn. Study of erosion of artificial neural network has been performed by several researchers by erosion-prone area prediction modeling with good resulting accuracy [6-7]. [8] Perform similar research, which is mapping landslide-prone area using artificial neural network with accuracy of prediction up to 82.92%. The weakness of artificial neural network is incomprehensible supporting theoretical evidence in selecting a combination of parameter value of ANN, including ANN structure selected to produce effective results, making ANN difficult to apply [8-9]. The current research tested several parameter combinations to produce good accuracy. The algorithm used for execution was backpropagation. There are many artificial neural network algorithms that can be used in the training process. Backpropagation algorithm in multilayer perceptron (MLP) network is one that is widely utilized for complex problems. The previous researcher [10] compare the guided algorithms: Backpropagation, Self-Organizing Mapping and Simulated Annealing using Landsat TM imagery in the classification of land cover conclude that the backpropagation method obtains the most accurate precision compared to other algorithms. Some other researchers also prove that backpropagation algorithm is reliable in the classification of spatial data with good accuracy [8,11-13].

Backpropagation algorithm uses error output to change the loading values backward. To fix this error, forward propagation phase must be performed in advance [14]. In addition to write algorithm selection, network operation is likewise influenced by the pattern of data input and the value of the network parameters, hence that the network can perform the function of generalization and memorization well. In this paper, the focus of study was comparing the combination of various input parameters of artificial neural network to get good classification results.

2. Method
The entire experiment was performed on Windows 8 (64-bit) operating system with i3 processor and 2 GB RAM. Artificial neural network training was performed on IDRISI using MLP (multilayer perceptron) tool. The training data consisted of 9 layers which are the factors of erosion control, i.e. erosivity (R), erodibility (K), length and slope (LS), land cover management (C), factors supporting land conservation practice (P) and four additional layers of remote sensing data, i.e. SPOT 5 multispectral imagery with four bands/channels that provided different spectral reflectance values for each land cover/land use. The neural network architecture in this modeling is as shown in Figure 1. Total data collected was 53 samples of which 30 samples were used as training data and 23 samples as test data for verification. The experiment conducted with 23 simulations by changing the values of network parameters, i.e. learning rate (LR), momentum (M), number of hidden layers (HL), number of iterations (I) and RMS Error (Table 1)

The training results were evaluated using an overall accuracy calculation. One that is widely used in accuracy calculation of the classification is by preparing classification error matrix. The correlation between known reference data and classification results can be compared with error matrix. Overall accuracy was computed by dividing the number of right pixels (the number of main diagonal) with the total number of pixels in the error matrix [15].
**Figure 1.** Architecture of artificial neural network for erosion (Band 1, band 2, band 3, band 4, band 5 = SPOT 5 satellite image), R = erosivity data image, K = erodibility data image, LS = length and slope steepness data image; c = vegetation cover management data image; P = supporting land conservation practice data image

**Table 1.** Parameters of network training structure

| Name   | HL | LR | M   | RMS  | I   | Name   | HL | LR | M   | RMS  | I   |
|--------|----|----|-----|------|-----|--------|----|----|-----|------|-----|
| ANN 1  | 1  | 0.001 | 0.5 | 0.0001 | 5000 | ANN 13 | 1  | 0.001 | 0.5 | 0.0001 | 25000 |
| ANN 2  | 1  | 0.001 | 0.5 | 0.0001 | 10000 | ANN 14 | 1  | 0.01 | 0.5 | 0.0001 | 15000 |
| ANN 3  | 1  | 0.001 | 0.5 | 0.0001 | 15000 | ANN 15 | 2  | 0.01 | 0.9  | 0.0001 | 10000 |
| ANN 4  | 2  | 0.001 | 0.5 | 0.0001 | 5000  | ANN 16 | 2  | 0.001 | 0.9  | 0.0001 | 15000 |
| ANN 5  | 2  | 0.001 | 0.5 | 0.0001 | 15000 | ANN 17 | 1  | 0.001 | 0.9  | 0.0001 | 5000  |
| ANN 6  | 1  | 0.001 | 0.9  | 0.0001 | 15000 | ANN 18 | 1  | 0.001 | 0.5  | 0.0001 | 20000 |
| ANN 7  | 2  | 0.4  | 0.5  | 0.0001 | 5000  | ANN 19 | 2  | 0.001 | 0.5  | 0.0001 | 20000 |
| ANN 8  | 1  | 0.01 | 0.5  | 0.001  | 5000  | ANN 20 | 1  | 0.001 | 0.9  | 0.0001 | 20000 |
| ANN 9  | 1  | 0.01 | 0.5  | 0.001  | 10000 | ANN 21 | 2  | 0.001 | 0.9  | 0.0001 | 20000 |
| ANN 10 | 2  | 0.01 | 0.5  | 0.001  | 5000  | ANN 22 | 2  | 0.001 | 0.9  | 0.0001 | 25000 |
| ANN 11 | 2  | 0.01 | 0.5  | 0.001  | 10000 | ANN 23 | 1  | 0.001 | 0.9  | 0.0001 | 30000 |
| ANN 12 | 2  | 0.01 | 0.4  | 0.001  | 10000 |        |    |    |     |      |     |
3. Experimental Results and Discussion

Table 2 shows the highest training accuracy achieved in this study was 99.32%, occurred in ANN 14 simulation with combination of network input parameters of 1 HL, LR 0.01; M 0.5; RMS 0.0001, and the number of iterations of 15000. The previous researcher [16] have proved the accuracy of training reaches 91.9% when the iterations at 15000. The experimental results showed there was a simulation with low accuracy and some simulations that cannot be executed by artificial neural network because the combination of network input parameters was not considered ideal in the simulations (Table 2)

| Name   | HL | LR  | M  | RMS    | I     | Overall accuracy (%) |
|--------|----|-----|----|--------|-------|----------------------|
| ANN 4  | 2  | 0.001 | 0.5 | 0.0001 | 5000  | error                |
| ANN 5  | 2  | 0.001 | 0.5 | 0.0001 | 15000 | 48.69                |
| ANN 7  | 2  | 0.4   | 0.5 | 0.0001 | 5000  | error                |
| ANN 15 | 2  | 0.01  | 0.9 | 0.001  | 10000 | error                |

Table 2 shows ANN 4 resulted in an error or “out of memory” whose results were not as expected. This condition occurred in the simulation with 2 HL, but cannot be said that only the hidden layer that affected on accuracy. The following are the effects of each network parameter against the level of accuracy. The first parameter, i.e. iteration, was the parameter that had great effect on accuracy. It can be seen from multiple simulations that had the same network parameter values, only differed in the value of iterations (Table 3)

| Name   | HL | LR  | M  | RMS    | Iteration | Overall accuracy (%) |
|--------|----|-----|----|--------|------------|----------------------|
| ANN 4  | 2  | 0.001 | 0.5 | 0.0001 | 5000       | error                |
| ANN 5  | 2  | 0.001 | 0.5 | 0.0001 | 15000      | 48.69                |
| ANN 8  | 1  | 0.01  | 0.5 | 0.001  | 5000       | 98.58                |
| ANN 9  | 1  | 0.01  | 0.5 | 0.001  | 10000      | 95.8                 |
| ANN 10 | 2  | 0.01  | 0.5 | 0.001  | 5000       | 98.37                |
| ANN 11 | 2  | 0.01  | 0.5 | 0.001  | 10000      | 99.05                |
| ANN 21 | 2  | 0.001 | 0.9 | 0.0001 | 20000      | 98.78                |
| ANN 22 | 2  | 0.001 | 0.9 | 0.0001 | 25000      | 98.06                |

Table 3 shows overtraining occurs in ANN 8 and ANN 9 simulations. ANN 8 with 5000 iterations had reached 98.58% accuracy, but when added by iterations in ANN 9 to 15000, the accuracy value was decreased. Basically, there was no provision or previous study mentioned the best number of iterations. The initial determination of the number of iterations will be a reference for the next simulation only when the first simulation is carried out. Thus, the thing to be considered when determining the number of iterations is the network is not stuck in overtraining, rather than what are the numbers of iteration that must be carried out in every training. Because the data pattern in each
case was different, the ideal treatment or artificial neural network architecture will also be different. Overall, from 23 simulations that were carried out, the best iteration was 15000 since it occurred in some simulations. The optimal number of iterations was also strongly affected by the combination with other parameters, such as momentum and RMS [13]. The simulations had been carried out with two different momentum values tested in different iterations (Table 4).

| Name   | HL | LR | M   | RMS    | I       | Overall accuracy (%) |
|--------|----|----|-----|--------|---------|-----------------------|
| ANN 1  | 1  | 0.001 | 0.5 | 0.0001 | 5000    | 81.81                  |
| ANN 2  | 1  | 0.001 | 0.5 | 0.0001 | 10000   | 89.8                  |
| ANN 3  | 1  | 0.001 | 0.5 | 0.0001 | 15000   | 94.58                 |
| ANN 13 | 1  | 0.001 | 0.5 | 0.0001 | 25000   | 95.48                 |
| ANN 18 | 1  | 0.001 | 0.5 | 0.0001 | 20000   | 96.16                 |
| ANN 17 | 1  | 0.001 | 0.9 | 0.0001 | 5000    | 96.34                 |
| ANN 6  | 1  | 0.001 | 0.9 | 0.0001 | 15000   | 98.51                 |
| ANN 20 | 1  | 0.001 | 0.9 | 0.0001 | 20000   | 98.6                  |
| ANN 23 | 1  | 0.001 | 0.9 | 0.0001 | 30000   | 99.23                 |

The changes of iteration carried out in the values of momentum in 0.5 and 0.9 showed M value = 0.5 higher accuracy by the addition of the number of iterations. The researcher performed the addition because of seeing the pattern of increased accuracy when the number of iterations added by the assumption of low accuracy because of the number of training was less (under training). An overview of accuracy changes at each iteration with the values of M = 0.5 and M = 0.9 is presented in Figure 2.

![Figure 2. The correlation between iteration, momentum, and accuracy](image)

The second parameter, i.e. learning rate, affected on the intensity of the training process. Learning rate becomes a problem when the number of training is too much, so that the system is not able to generalize well. Learning rate that is too high requires a longer time for continuous training, so that
optimal convergence is not achieved [17]. Lower learning rate guarantees a decrease in gradient, but can increase the number of iterations. In the simulations carried out in the study, the most optimal learning rates producing the best accuracy was 0.001 and 0.01, as on previous research conducted by [13]. The correlation with accuracy and time was inversely related. The lower the value of learning rate, the higher the accuracy, and the longer the computation time. Otherwise, the higher the value of learning rate, the lower the accuracy, and the faster the computation time. The value of learning rate that was higher had been tested in the simulation conducted at ANN 7, i.e. 0.4 (Table 2), but the results of training cannot be executed. In contrast to those produced by [16], learning rate of 0.4 produces an accuracy of 75%. Generally, the value of LR been ranging from 0.001 to 1 [18]. The study had been attempted with LR 0.001; 0.01 was quite optimal to use. Both values have also been proven by [13]. His study can yield the highest accuracy, but also can yield low accuracy if other input parameters are not correct.

The third parameter, i.e. hidden layer, functions as a place where algorithm improves the load of input before comes to the output layer. ANN has knowledge rules that are difficult to formulate as seeing how much each of the parameters affect the accuracy, including hidden layer. Some simulations with 1 HL had very good accuracy, but contrarily, some simulations with 2 HL had better accuracy. Some of the following simulation shows the difference in accuracy between 1 HL and 2 HL (Table 5)

| Name  | HL | LR   | M   | RMS   | I   | Overall accuracy (%) |
|-------|----|------|-----|-------|-----|----------------------|
| ANN 1 | 1  | 0.001| 0.5 | 0.0001| 5000| 81.81                |
| ANN 4 | 2  | 0.001| 0.5 | 0.0001| 5000| error                |
| ANN 3 | 1  | 0.001| 0.5 | 0.0001| 15000| 94.58               |
| ANN 5 | 2  | 0.001| 0.5 | 0.0001| 15000| 48.69               |
| ANN 8 | 1  | 0.01 | 0.5 | 0.001 | 5000 | 98.58               |
| ANN 10| 2  | 0.01 | 0.5 | 0.001 | 5000 | 98.37               |
| ANN 9 | 1  | 0.01 | 0.5 | 0.001 | 10000| 95.8                |
| ANN 11| 2  | 0.01 | 0.5 | 0.001 | 10000| 99.05               |
| ANN 18| 1  | 0.001| 0.5 | 0.0001| 20000| 96.16               |
| ANN 19| 2  | 0.001| 0.5 | 0.0001| 20000| 74.72               |
| ANN 20| 1  | 0.001| 0.9 | 0.0001| 20000| 98.6                |
| ANN 21| 2  | 0.001| 0.9 | 0.0001| 20000| 98.78               |

Based on Table 5 above, the combination of parameters of LR 0.001; M 0.5; RMS 0.0001 would be optimal with 1 HL although changes were made to the number of iterations as in simulations ANN 1 and ANN 4, ANN 3 and ANN 5, ANN 18 and ANN 19. When the momentum parameter and the value of learning rate were changed, the resulting accuracies on some combinations with 2 HL were optimal, and in other combinations with 1 HL were sufficient. These conditions occurred in the simulations ANN 9 and ANN 11, ANN 20, and ANN 21. Overall, the best accuracy occurred with 1 HL. Some simulations with 2 HL achieved the highest accuracy as in ANN 11 and ANN 21, but accuracy with 1 HL on the same simulations and parameters had reached an accuracy of > 90%. So, it can be said that one hidden layer was sufficient to solve the problems with the characteristics of the data in this study.
[14] supports the above facts that 1 HL has been already quite optimized to use, because the more the number of HL, the lower the computing process; even in some cases, the computer cannot process them as in ANN 4 and ANN 5.

The fourth parameter, i.e. **momentum**, functions to accelerate the convergence of the algorithm by forcing the load change process to continue to move, so that not being trapped in a local minimum, resulting in the training process to be faster. However, the training time in this study was not recorded since not been the focus of study. The difference in accuracy with different momentum values is presented in Table 6.

| Name    | HL | LR   | M   | RMS         | I    | Overall accuracy (%) |
|---------|----|------|-----|-------------|------|----------------------|
| ANN 1   | 1  | 0.001| 0.5 | 0.0001      | 5000 | 81.81                |
| ANN 17  | 1  | 0.001| 0.9 | 0.0001      | 5000 | 96.34                |
| ANN 3   | 1  | 0.001| 0.5 | 0.0001      | 15000| 94.58                |
| ANN 6   | 1  | 0.001| 0.9 | 0.0001      | 15000| 98.51                |
| ANN 11  | 2  | 0.01 | 0.5 | 0.001       | 10000| 99.05                |
| ANN 12  | 2  | 0.01 | 0.4 | 0.001       | 10000| 99.23                |
| ANN 18  | 1  | 0.001| 0.5 | 0.0001      | 20000| 96.16                |
| ANN 20  | 1  | 0.001| 0.9 | 0.0001      | 20000| 98.6                 |
| ANN 19  | 2  | 0.001| 0.5 | 0.0001      | 20000| 74.72                |
| ANN 21  | 2  | 0.001| 0.9 | 0.0001      | 20000| 98.78                |

Based on the observation on the results of simulations, the high value of learning rate required lower momentum, otherwise low value of learning rate required higher value of momentum to achieve optimal effectiveness and convergence during the training process. It can be seen from the simulation results of ANN 6, ANN 20, and ANN 21, in which the momentum, value of 0.9 in learning rate value of 0.001 was more optimal than the simulation using a momentum value of 0.5. While [11] give a rather good accuracy on the momentum value of 0.5.

The fifth parameter that affected the network was **root mean square (RMS) error**. RMS is the root mean square that occurs between the network output and target output. This study used two RMS values in simulations, i.e. 0.001 and 0.0001. From the test results, the highest accuracy on each combination occurred in RMS value of 0.0001. The analysis results of the entire simulation performed showed both the RMS values (0.001 and 0.0001) likely to yield better accuracy if the values of other network parameters (learning rate, hidden layers, iteration) were right through trial-and-error. The comparison of the accuracy, resulting in both RMS values is presented in the following Table 7.

| Name    | HL | LR | M  | RMS         | I    | Overall accuracy (%) |
|---------|----|----|----|-------------|------|----------------------|
| ANN 9   | 1  | 0.01| 0.5| 0.0001      | 10000| 95.80                |
| ANN 14  | 1  | 0.01| 0.5| 0.0001      | 15000| 99.32                |
The number of iterations has a correlation with the value of the RMS. The small RMS error requires many iterations. Otherwise, the great RMS error requires fewer iterations, as proven by [13]. The same conclusion obtained in this study as described in the graphic of the results of simulations ANN 1 and ANN 2 (Figure 3).

![Figure 3. The Correlation between RMS Error and Iteration (a)ANN 1; (b) ANN 2](image)

Test results on 23 simulations showed the highest accuracy ANN 14 simulation with overall accuracy reaching 99.32%. This resulting accuracy is higher than that produced by [19] in the land use classification using artificial neural networks which resulted in the training accuracy by 95%. An image of classification of ANN 14 is shown in Figure 4 where distribution of erosion class in the research site was dominated by the following erosion classes, namely very severe erosion (34.22%) spreading across most of the area of Kokap Sub-district, Girimulyo Sub-district, and some of the area of Pengasih Sub-district; severe erosion (20.75%) spreading all over Panjatan Sub-district, Pengasih Sub-district, and Nanggulan Sub-district; moderate erosion (29.78%) spreading across Pengasih Sub-district, some of the area in Wates Sub-district, and Panjatan Sub-district; as well as slight erosion (10.81%) and very slight erosion (4.46%) spreading all over Temon Sub-district and Wates Sub-district. The percentages of distribution for each erosion class are shown in the histogram presented in Figure 5.
**Figure 4.** Map of spatial distribution of erosion (Analysis, 2016)

**Figure 5.** The Histogram for the Distribution of Erosion Classes According to the Results of the ANN 14 (Analysis, 2016)
4. Conclusion
The artificial neural network is effectively used for the classification of spatial data, although trained by the small number of samples. The complexity of data input can be understood well by the network while training with the accuracy of 99.32%. The most influential parameter in this study was the number of iterations although generally, the right combination among network parameters was also an important concern. The parameter of hidden layer did not have a significant effect on the level of accuracy.

Acknowledgement
The authors would like to thank National Institute of Aeronautics and Space, Indonesia, Meteorological Climate and Geophysics Agency for providing the data. Financial support from the Ministry of Research, Technology and Higher Education of the Republic of Indonesia, which has provided postgraduate scholarship in Universitas Gadjah Mada, Yogyakarta and funding this Doctoral Dissertation Skim.

References

[1]. Arsyad S 1989 Konservasi Tanah dan Air Penerbit IPB Press Bogor
[2]. Bouaziz M, Leidig M, Gloaguen R 2011 Optimal Parameter Selection for Qualitative Regional Erosion Risk Monitoring: A Remote Sensing Study of SE Ethiopia. Geoscience Frontiers 2 (2) pp 237-245
[3]. Morgan R P C 1995 Soil Erosion and Conservation, 2nd Edition. Longman Group, Ltd., London, 198 p.
[4]. Parveen R, Kumar U 2012 Integrated Approach of Universal Soil Loss Equation (USLE) and Geographical Information System (GIS) for Soil Loss Risk Assessment in Upper South Koel Basin, Kharkhand. J.Geographic Information System 4 pp 588-596
[5]. Melchiorre C, Matteucci A, Azzoni A, Zanchi 2008 Artificial Neural Networks and Cluster Analysis in Landslide Susceptibility Zonation Geomorphology 94 pp 379-400
[6]. De la Rosa D, de la Mayol, F Lozano S 1999 An Expert System/Neural Network Model (impelERO) for Evaluating Agricultural Soil Erosion in Andalucia Region, Southern Spain. Agriculture, Ecosystem and Environment 73 pp 211-226
[7]. Kim M, Gilley J E 2008 Artificial Neural Network Estimation of Soil Erosion and Nutrient Concentrations in Runoff From Land Application Areas Comput Electron Agric. doi://10.1016/j.compag.2008.05.021
[8]. Pradhan B, Lee S, Buchroitner M F 2010 A GIS-Based Backpropagation Neural Network Model And Its Cross-Application and Validation for Landslide for Susceptibility Analyses. Computers Environment and Urban Systems 34 pp 216-235
[9]. Liu Q, Wu G, Chen J, Zhou G 2012 Interpretation Artificial Neural Network in Remote Sensing Image Classification. IEEE
[10]. Yuan H, C F Van Der Wiele, S Khoram 2009 An Automatad Artificial Neural Network System for Land Use/Land Cover Classification from Landsat TM Imagery Remote Sens. 1 (3) pp 243-265
[11]. Song H, Xu, Ma Y, Li G 2013 Classification of ETM+ Remote Sensing Image Based on Hybrid Algorithm of Genetic Algorithm and Backpropagation Neural Network Hindawi Publishing Corporation, Mathematical Problems Engineering
[12]. Conforti M, Pascale S, Robustelli G, Sdao F 2014 Evaluation of Prediction Capability of The Artificial Neural Networks for Mapping Landslide Susceptibility in The Turbolo River Catchment (northern Calabria, Italy) Catena 113 pp 236-250
[13]. Arif N, Danoedoro P 2013 An Analyze of Backpropagation Neural Network in The Identification of Critical Land Based on Alos Imagery Proc. Int. Conf. on 34th Asian Conference on Remote Sensing, Volume 1 pp 589-593
[14]. Fausett L 1994 Fundamentals Of Neural Networks: Architectures, Algorithms, and Applications, Prentice-Hall, New Jersey
[15]. Congalton R G, Green K 2009 Assessing the Accuracy of Remotely Sensed Data: Principles and Practices CRC Press
[16]. Chen Y R, Chen J W, Hsieh S C, Ni P N 2009 The Application of Remote Sensing Technology to The Interpretation of LU for Rainfall – Induced Landslide Based on Genetic Algorithm and Artificial Neural Network IEEE Journal of Selected Topics in Applied Earth Observations and remote Sensing, Vol 2 No 2 pp 87-95
[17]. Wilson W R, Martinez T R 2001 The Need for Small Learning Rate on Large Problems. Proceedings of The International Journal Conference on Netherlands pp 115 – 119
[18]. Purnomo M H, Kurniawan A 2006 Supervised Neural Networks, Graha Ilmu, Yogyakarta, 2006 (Text in Indonesia)
[19]. Chen Y R, Chen J W, Hsieh S C, Ni P N 2013 The Application of Remote Sensing Technology to The Interpretation of LU for Rainfall – Induced Landslide Based on Genetic Algorithm and Artificial Neural Network IEEE Earth Observations and remote Sensing, Vol 2 (2) pp 87-95