Error Rate Testing of Training Algorithm in Back Propagation Network

Hindayati Mustafidah, Suwarsito

Abstract— ***Artificial Neural Network (ANN), especially back propagation method has been widely applied to help solve problems in many areas of life, eg for the purposes of forecasting, diagnostics, and pattern recognition. An important part at ANN in determining the performance of the network is training algorithm used. Because there are 12 training algorithms that can be used at back propagation method, of course, it's needed to be selected the most optimal algorithm in order to obtain the best results. Training algorithm performance is said optimal in providing solutions can be seen from the error generated. The smaller the error is generated, the more optimal performance of the algorithm. In this study, testing to get the training algorithm has the smallest error rate of 12 existing algorithms. Testing begins with the preparation of a computer program modules using MATLAB programming language to get the error value of the network output for each training algorithm. Each program for each training algorithm executed 20 times. Furthermore, the error of the network output was tested using analysis of variance with an alpha level of 5% to get a training algorithm which has the smallest error rate. The conclusion of the test results is that the training algorithm "trainlm" has the smallest error with the network parameters for the target error = 0.001 (10^{-3}), the maximum epoch = 10000, learning rate (lr) = 0.01, and 5 neuron input data with 1 neuron output.***

Keywords: error rate, training algorithm, back propagation, network parameters

I. INTRODUCTION

Artificial Neural Network (ANN) is an information processing system that has characteristics similar to biological neural network, which was formed as a generalization of mathematical models of biological neural networks [1]. ANN is the ideal solution for problems that can't be formulated easily using the algorithm [2]. This system has the capability of storing knowledge based on experience and to make that knowledge to be useful. ANN created as a generalization mathematical model of human understanding (human cognition). As a model, ANN uses approach in conducting reasoning in solving problems that can be either functional or through random search. Backpropagation is a supervised learning algorithm and the most widely used, with more than one layer (multi-layer) to change the weights are connected with neurons that exist in the hidden layer. Backpropagation algorithm using the error output to change the value of the weights in the backward direction. To get this error, advanced propagation phase (forward propagation) must be done first. At the time of forward propagation, neurons will be activated by using an activation function which can be differentiated.

Revised Version Manuscript Received on August 04, 2015.

Hindayati Mustafidah, Department of Informatic Engineering, University Muhammadiyah of Purwokerto, Central Java, 53182, Indonesia.

Suwarsito, Department of Geography Education, University Muhammadiyah of Purwokerto, Central Java 53182, Indonesia.

This activation function must meet several requirements, namely: continuous, differentiable easily, and is a function that does not go down [1]. These functions such as:

- Sigmoid binary that has range (0, 1) with equation 1 as follows:

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

where $f'(x) = \sigma f(x)[1 - f(x)]$

Binary sigmoid function graph is presented in Figure 1.

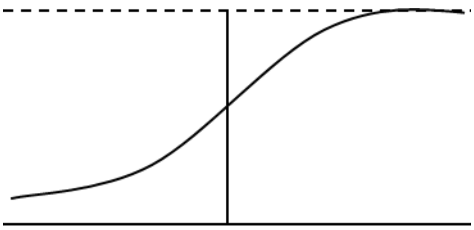

Fig.1: Graph Function Sigmoid

- tansig with membership function as in equation 2 or 3.

$$y = f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad (2)$$

or

$$y = f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \quad (3)$$

where $f'(x) = [1 + f(x)][1 - f(x)]$

- purelin

$y = f(x) = x$

where $f'(x) = 1$

Some topics that discussed on ANN are more on the application and development of algorithms of learning/training. There are several algorithms contained in the ANN training, including the Fletcher-Reeves Update (traincgf), Polak-Ribiere (traincgp), Powell-Beale restarts (traincgb), and Scaled Conjugate Gradient (trainscg), Gradient Descent with Adaptive Learning Rate (traingda), Gradient Descent with Momentum and Adaptive Learning Rate (traingdx), Resilent backpropagation (trainrp), BFGS (trainbfg), One Step secant (trainoss), Levenberg-Marquardt (trainlm) belonging to the back propagation method [3]. Error calculation is a measurement of how the network can learn well so when compared with the new pattern will be easily recognizable. Error at the output of the network is the difference between actual output (current output) and the desired output [4]. The resulting difference between them is usually determined in a manner calculated using an equation. In this study, the error is calculated using the MSE (Mean Squared Error). MSE is a function of network performance based on the average of the squared error. The equation used is as in equation 4 below.

$$MSE = \frac{\sum_p \sum_j (\tau_{jp} - x_{jp})^2}{n_p n_o} \quad (4)$$

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

where T_{jp} = the output value of the neural network
X_{jp} = the desired target value for each output
n_p = the total pattern
n_o = the number of outputs

Some of the training algorithm has been independently tested and applied in a case to solve the problem, but has not been thoroughly tested among existing algorithms. Training algorithm raingd has been applied to solve the problems predicting student achievement at Informatics Engineering Program in University of Muhammadiyah Purwokerto based on the values of the subjects tested in the national examination while in high school [5] and produce error rates for 0.0664179 of a target error of 0.05. Traincgf, raincgp, traincgb, and trainscg training algorithms have been tested their accuracies in the case solving of student achievement prediction [6] and the conclusion that the algorithm of the fourth with 5% alpha did not differ significantly in accuracy. Several algorithms have not been tested for their optimality level (produce the smallest error rate) and as a follow up of the study of [6] and [7], it is necessary to know the level of optimality other training algorithms to solve the problems. While the level of accuracy of the data pattern recognition of some algorithms that are traingda, traingdx, trainrp, trainbfg, trainoss, and trainlm have been tested by [8] and produce information that the confidence level of 95% showed that the algorithm trainlm is a most meticulous algorithm with average error 0.0063. Meanwhile [9] also has implemented ANN to predict the level of validity problems by using training algorithm trainlm and generating the data pattern recognition to have a match at 86.54% with an error rate of 0.00063347. Therefore, it is necessary to test the accuracy level of training algorithms in terms of network error generated for the case of random both input and output data.

II. METHOD

This study is a mixed method that is the research development with quantitative and qualitative testing (using a statistical test). As a research development, this study begins with a computer program developed using MATLAB programming language to run 12 network training algorithms that are trainbfg, traincgb, traincgf, traincgp, traingd, traingda, traingdm, traingdx, trainlm, trainoss, trainrp, and trainscg. Overall, the steps taken in this study are as follows:

A. Determine Input and Output Data Networks

Network input data is in the form of random data network with 5 neurons structure, while the output data consists of one neuron network.

B. Develop a Program Code

Program code is developed to generate data random of input and target.

C. Build ANN Structure

In this structure involved research variables are:

- Control Variables: max epoch (= 10,000) and the error target (= 10^{-3}).
- Independent variables: data input and learning rate (lr).
- Dependent variables (factors): training algorithms
- Output: error rates generated by each training algorithm.

D. Develop Program Codes for Each Training Algorithm

Codes for training algorithms are developed according to the structure and parameters of such networks are built in step C.

E. Conduct Network Training Process to Produce Output

Training process is done in 20 repetitions for each training algorithm.

F. Perform Statistical Test of the Output Produced and Conclude It

Tests conducted to determine the optimization of training algorithms by comparing the level of errors that occur with the following steps [10]:

1) *determine the hypothesis*

   H_0: $\mu_1 = \mu_2 = \ldots = \mu_k$ (all training algorithm is homogeneous / has the same properties)

   H_1: there are several training algorithms that are not homogeneous / do not have the same properties
2) *determine the value of alpha (α)*
3) *determine the test instrument*

   Test instrument that is used in this case is the F test (equation 5):

$$F_{calc} = \frac{MST}{MSE} \sim F_{k-1, N-k} \quad (5)$$

   where MST = *Mean Square of Treatment*
   MSE = *Mean Square of Error*
4) *conclude*

   Conclusions drawn by the significant value gained (sig.) with the provisions of H_0 is rejected if the value of sig<α.

Design of ANN program structure presented in Figure 2.

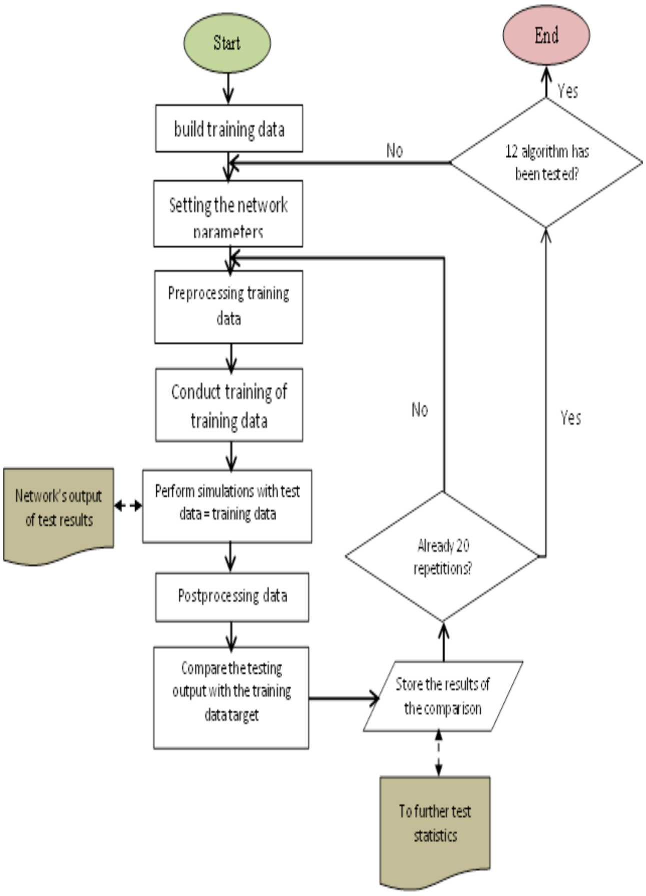

Fig. 2: The flow of ANN program development

III. RESULT AND DISCUSSION

A. ANN Structure

ANN structure used is presented in Figure 3.

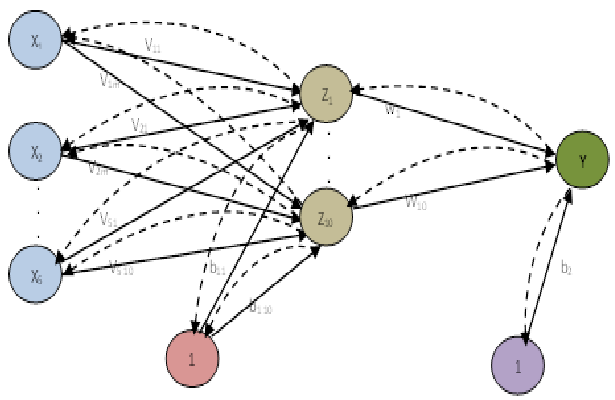

Fig. 3: Backpropagation Network Architecture

In Figure 3 the network consists of 5 units of neurons in the input layer, namely x_1, x_2, ..., x_5, one hidden layer with 10 neurons that z_1, ..., z_{10}, and 1 unit of neurons in the output layer is Y. Weight that connect neurons in the input layer to the hidden layer neurons are v_{11} (x_1 linking weight to z_1), v_{12} (weights connecting z_2 to x_1), ..., v_{1_10} (weights connecting x_1 to z_{10}), v_{21} (weights which connects z_1 to x_2), v_{22} (weights connecting z_2 to x_2), ..., v_{2_10} (weighting x_2 connects to z_{10}), ..., v_{5_10} (weighting x_5 connect to z_{10}). The weighting is symbolized by v_{ij} (ie weight connecting i-th input neuron to neuron j in the hidden layer). The weight bias that led to the first and second neuron in the hidden layer is b_{11} (bias weight that connects to z_1), ..., b_{1_10} (bias weight which connects to z_{10}). Weights that connect neurons in the hidden layer is z_1, ..., z_{10} by neurons in the output layer (Y) is w_1, ..., w_{10}. While the weight is the weight bias b_2 leading bias to the neurons in the output layer. Activation function which is used between the input layer and the hidden layer and the hidden layer to the output layer is the activation function which can be differentiated namely tansig and purelin.

B. Network Training

Network trained using the input data and the target that is worth random. Data input consists of 5 data variation (5 neurons) with 5 data patterns with each pattern paired with one target data (Table 1).

Table 1: Data input with the number of neurons = 5

X_1	X_2	X_3	X_4	X_5	Y
9.5013	7.6210	6.1543	4.0571	0.5789	2.0277
2.3114	4.5647	7.9194	9.3547	3.5287	1.9872
6.0684	0.1850	9.2181	9.1690	8.1317	6.0379
4.8598	8.2141	7.3821	4.1027	0.0986	2.7219
8.9130	4.4470	1.7627	8.9365	1.3889	1.9881

C. ANN Program Code

Program code created for each training algorithm. The maximum control variables of epoch (= 10000) and the target error (= 10^{-3}). While the independent variable in the form of input data network with 5 variations of data patterns, and learning rate (lr) = 0.01, respectively. Program code developed to test the level of error in 12 training algorithms as the dependent variable or factor variables. Each algorithm is executed 20 times. The output of the program is the level of error (MSE) that is generated at lr = 0.01 of twelfth training algorithms (presented in Table 2).

Table 2: MSE at the level lr = 0:01

No.	trainbfg	traincgb	traincgf	traincgp	traingd	traingda
1	0.0781685	0.000310694	0.000977715	0.000774416	0.000965436	0.000938196
2	0.000879153	0.000887382	0.000911287	0.000409749	0.000950124	0.000952874
3	0.000389766	8.42209e-005	3.9733e-005	7.17637e-005	0.000987908	0.000916735
4	0.000421823	0.000581543	0.000477211	0.00056505	0.000961722	0.000959623
5	2.458	0.000831864	0.000268732	0.000899921	0.000986689	0.000874886
6	2.458	0.000677702	0.000932816	0.000941192	0.000964307	0.000827337
7	0.0781685	0.000788396	0.00046224	0.000576934	0.000950927	0.000950306

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

8	0.000280615	0.000179958	0.000919173	0.000391876	0.00094071	0.000985391
9	0.0481914	0.0539822	0.000451758	7.82838e-005	0.000996653	0.000993256
10	0.0481914	0.000364023	0.0481944	0.000232553	0.000983699	0.000948084
11	4.43199e-005	5.00224e-005	0.000811175	0.000877537	0.000978479	0.000956031
12	0.000436739	0.000700925	0.000936189	0.000984303	0.000958102	0.000885283
13	0.0781685	0.000853109	0.000659036	0.000389815	0.000963321	0.000971769
14	2.458	0.000689615	0.000423028	0.0004295	0.000969133	0.000843659
15	0.000672696	0.000800053	0.000347741	0.000534345	0.000982076	0.000972777
16	0.000750594	0.00085553	0.000721401	0.000169477	0.000942071	0.000924146
17	2.458	0.000722605	0.000940585	0.000171294	0.000947663	0.000999746
18	0.000654728	0.000906544	0.0003173	0.000364043	0.000994409	0.000948903
19	0.000327463	8.95564e-005	0.000268001	0.00063078	0.000972943	0.00091782
20	0.0539784	0.000483212	0.000848179	0.000576648	0.000968354	0.000991755

Table 2: (continued)

No.	traingdm	traingdx	trainlm	trainoss	trainrp	trainscg
1	0.000990477	0.000918051	2.76285e-006	2.458	0.000970461	0.000942185
2	0.000986231	0.000903277	1.91709e-005	0.000955535	0.000827962	0.000706939
3	0.000981085	0.000818023	0.000157904	0.00095813	0.000670791	0.000229741
4	0.000938916	0.000984472	0.000719061	0.000528402	0.000366767	0.00072544
5	0.000991027	0.000749829	2.34459e-006	2.458	0.000574537	0.000997388
6	0.000993436	0.000836939	0.000120145	0.000956981	0.000837413	0.000688001
7	0.000956134	0.000963453	5.45649e-006	2.458	0.000873104	0.000384876
8	0.000994875	0.000974023	0.000153304	0.000646223	0.000784764	0.00093666
9	0.000990672	0.000968763	0.000516044	0.000908489	0.000591384	0.000350462
10	0.000990539	0.000881465	0.000231007	0.0539784	0.000480217	0.00072921
11	0.000983341	0.000855156	0.000180418	2.458	0.000764007	0.000597558
12	0.000954641	0.000970021	0.000660536	0.000685945	0.000407189	0.000798862
13	0.000943529	0.000946187	0.000103428	0.00099052	0.000822803	0.000472698
14	0.000993679	0.000990289	0.000351304	2.458	0.000867431	0.000977186
15	0.000960715	0.000977001	0.000212775	0.000933286	0.000874552	0.000673188
16	0.000947285	0.000948982	3.97761e-005	0.000767107	0.00065295	0.000616146
17	0.00098698	0.000686511	0.000287332	0.000516164	0.000979903	0.00099715
18	0.000919398	0.00081974	2.91726e-006	0.0781685	0.000445106	0.000538651
19	0.000997221	0.000944422	0.000122912	0.00072297	7.80285e-005	0.000592311
20	0.00097243	0.000989885	8.51178e-005	0.000758746	0.000878979	0.000986976

D. Testing Error Rate for Training Algorithm

Under the program output generated, then performed a statistical analysis (Analysis of Variance/ANOVA) to get the training algorithm to get the smallest error rate at the level lr = 0.01. The test results are presented in Table 3, Table 4, Table 5, and Table 6.

Table 3: ANOVA Test of 12 Training Algorithms MSE

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	15.353	11	1.396	5.724	.000
Within Groups	55.593	228	.244		
Total	70.947	239			

Table 4: Duncan Test of 12 Training Algorithms

MSE	algorithm	N	Subset for alpha = 0.05	
			1	2
Duncan[a]	trainlm	20	.0001986858	
	traincgp	20	.0005034740	
	trainrp	20	.0006874174	
	trainscg	20	.0006970814	
	traingdx	20	.0009063244	
	traingda	20	.0009379289	
	traingd	20	.0009682363	
	traingdm	20	.0009736305	
	traincgf	20	.0029953850	
	traincgb	20	.0032419577	
	trainoss	20		.6216237699
	trainbfg	20		.7326835139
	Sig.		.988	.478

Means for groups in homogeneous subsets are displayed.

a. Uses Harmonic Mean Sample Size = 20.000.

Based on Table 3, it appears that the sig. = 0.000 which means< α (= 0.05). It means that there is training algorithm that has MSE (error) different among twelve training algorithms. To know which training algorithm that have different error, used Duncan test and showed that the algorithm trainoss and

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

trainbfg have the same error level and far greater than the 10 other training algorithms (Table 4). Therefore, ANOVA test continued for 10 training algorithm rather than trainoss and trainbfg (Table 5). Because the data error is too small, so they're normalized using the square root (Table 6).

Table 5: ANOVA Test of 10 Training Algorithms

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	.092	9	.010	6.958	.000
Within Groups	.280	190	.001		
Total	.373	199			

Table 6. Duncan Test of 10 Training Algorithms

Duncan **norm1**

algorithm	N	Subset for alpha = 0.05		
		1	2	3
trainlm	20	.1024		
traincgp	20		.1441	
trainrp	20		.1592	.1592
trainscg	20		.1606	.1606
traincgb	20		.1647	.1647
traincgf	20		.1685	.1685
traingdx	20			.1734
traingda	20			.1750
traingd	20			.1764
traingdm	20			.1766
Sig.		1.000	.074	.229

ANOVA test results obtained sig. = 0.000 means tenth algorithms do not all have the same error level (Table 5). After Duncan test, produced the conclusion that the training algorithm trainlm has the smallest error compared with 9 other training algorithms (Table 6).

IV. CONCLUSIONS

Tests for training algorithm in backpropagation conducted using network parameters such as the target error = 0.001 (10^{-3}), the maximum epoch = 10000, learning rate (lr) = 0.01, with 5 input neurons and one output neuron. Based on the test results of 12 training algorithms, the algorithm trainlm having the smallest error with the level of $\alpha = 5\%$ and provides error of 0.0001986858. Thus trainlm algorithm can be considered as a training algorithm in backpropagation for application in solving problems.

ACKNOWLEDGMENT

Authors thank to DP2M - The Ministry of Research, Technology, and Directorate General of Higher Education (DIKTI) through KOPERTIS Region VI who has provided funds in the implementation of this research.

REFERENCES

[1] J.J. Siang, 2009, *Neural networks and programming Using MATLAB*, ANDI, Yogyakarta.
[2] M.T. Jones, 2008, *Artificial IntelligenceA Systems Approach*. Infinity Science Press LLC, New Delhi.
[3] S. Kusumadewi , S. Hartati, 2006, *Integration of Fuzzy Systems and Neural Networks*, Graha Ilmu, Yogyakarta.
[4] S. Kusumadewi, 2004, *Develop Artificial Neural Networks Using MATLAB and EXCEL LINK*, Graha Ilmu, Yogyakarta.
[5] Harjono, D. Aryanto, 2009, "Application of Artificial Neural Networks to Predict Student Achievement Study", *SAINTEK ISSN1411-2558, Vol. 5 No. 2*.
[6] H. Mustafidah, D. Aryanto, D.K. Hakim, 2013, "Optimization Test of Algorithm Conjugate Gradient Training on Artificial Neural Networks", *Proceeding SENATEK*, ISBN: 978-602-14355-0-2 pp. B.9-1. 21st September 2013.
[7] H. Mustafidah, D.K. Hakim, S. Sugiyanto, 2013, "Optimization Level of Training Algorithm on Artificial Neural Networks (Case Study: Student Achievement Prediction)", *JUITA* ISSN: 2086-9398 Vol. II No. 3, May 2013, pp. 159 – 166.
[8] F. Wibowo, S. Sugiyanto, H. Mustafidah, 2013, "Data Pattern Recognition Accuracy Level on Neural Network Improved Training Algorithm Method in Batch Mode", *JUITA*, ISSN 2086-9398, Vol. II No. 4, November 2013, pp. 259 – 264.
[9] H. Mustafidah, S. Hartati, R. Wardoyo, A. Harjoko, 2013, "Prediction of Test Items Validity Using Artificial Neural Network", *Proceeding International Conference on Education, Technology, and Science (NETS) 2013*, "Improving The Quality Of Education To Face The Impact Of Technology". December 28th, 2013. University Muhammadiyah of Purwokerto.
[10] T. Taniredja, H. Mustafidah, 2011, "*Quantitative Research (an Introduction)"*, Alfabeta, Bandung.

Hindayati Mustafidah, S.Si., M.Kom. Master's degree in computer science at University Gadjah Mada (UGM) graduating with honors in 2002, Lecturer at the University Muhammadiyah of Purwokerto in Informatics Engineering. Several times as presenter in national and international seminars; more than 20 papers has been published in national and international journals; many books have been published. The interested area of research is the field of intelligent systems, soft computing; and incorporated in the profession organization of Indonesian Computer Electronics and Instrumentation Support Society (IndoCEISS).

Suwarsito, S.Pi., M.Si. Master of fisheries at Bogor Agricultural University (IPB); Lecturer at the University Muhammadiyah of Purwokerto in Geography Education; the interested area of research are educational research fields and the science of waters. More than 20 papers has been published in national and international journal.