2009 IEEE International Advance Computing Conference (IACC 2009)
Patiala, India, 6–7 March 2009

Performance Comparison of Multi-layer Perceptron (Back Propagation, Delta Rule and Perceptron) algorithms in Neural Networks

Mutasem khalil Alsmadi
Dept. Information science
University Kebangsaan Malaysia
Kuala Lumpur, Malaysia
Mutasem_2004@yahoo.com

Khairuddin Bin Omar
Dept. System Science and Management
University Kebangsaan Malaysia
Kuala Lumpur, Malaysia
ko@ftsm.ukm.my

Shahrul Azman Noah
Dept. System Science and Management
University Kebangsaan Malaysia
Kuala Lumpur, Malaysia
samn@ftsm.ukm..my

Ibrahim Almarashdah
Dept. Industrial Computing
University Kebangsaan Malaysia
Kuala Lumpur, Malaysia
ibrahim_78us@yahoo.com

***Abstract*---A multilayer perceptron is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate output. It is a modification of the standard linear perceptron in that it uses three or more layers of neurons (nodes) with nonlinear activation functions, and is more powerful than the perceptron in that it can distinguish data that is not linearly separable, or separable by a hyper plane. MLP networks are general-purpose, flexible, nonlinear models consisting of a number of units organised into multiple layers. The complexity of the MLP network can be changed by varying the number of layers and the number of units in each layer. Given enough hidden units and enough data, it has been shown that MLPs can approximate virtually any function to any desired accuracy. This paper presents the performance comparison between Multi-layer Perceptron (Back Propagation, Delta Rule and Perceptron). Perceptron is a steepest descent type algorithm that normally has slow convergence rate and the search for the global minimum often becomes trapped at poor local minima. The current study investigates the performance of three algorithms to train MLP networks. Its was found that the Perceptron algorithm are much better than others algorithms.**

Keywords: Back propagation, perceptron , delta rule learning, classification .

I) INTRODUCTION

Recognition and cataloging are the vital facets in this up-to-the-minute era of research & development, hence exploiting the accessible techniques in Artificial Intelligence (AI) and Data Mining (DM) to achieve optimal production levels, examination procedures, and enhancing methodologies in most fields principally in the agricultural domain.

Artificial neural networks are defined as computational models of nervous system. Significantly natural organisms do not only possess nervous system; in fact they also evolve genetic information stored in the nucleus of their cells (genotype). Furthermore, the nervous system as a whole is part of the phenotype which is derived from the genotype through a specific development process. The information specified in the genotype determines assorted aspects of the nervous system which are expressed as innate behavioral tendencies and predispositions to learn [7], acknowledges that when neural networks are viewed in the broader biological context of Artificial Life, they tend to be accompanied by genotypes and to become members of budding populations of networks in which genotypes are inherited from parents to offspring. Many researchers such as Holland, Schwefel, and Koza, have stated that Artificial Neural Networks are evolved by the utilization of evolutionary algorithms.

The Perceptron: is a type of artificial neural network invented in 1957 at the Cornell Aeronautical Laboratory by Frank Rosenblatt. It can be seen as the simplest kind of feed forward neural network: a linear classifier [3].The learning algorithm is the same across all neurons; therefore a pattern that follows is applied to a single neuron in isolation.

II) BACK PROPAGATION

Back propagation algorithm was initially formulated by Werbos (1974) which was later modified by Rumelhart and McClelland (1986).Back propagation is the steepest decent type algorithm where the weight connection between the j-th neuron of the (k-1)-th layer and the i-th neuron of the k-th layer are respectively updated according to the following equation:

$$w_{ij}^k(t) = w_{ij}^k(t-1) + \Delta w_{ij}^k(t)$$
$$b_i^k(t) = b_i^k(t-1) + \Delta b_i^k(t) \quad (1)$$

with the increment $\Delta w_{ij}^k(t)$ and $\Delta b_i^k(t)$ given by

$$\Delta w_{ij}^k(t) = \eta_w \rho_i^k(t) v_j^{k-1}(t) + \alpha_w \Delta w_{ij}^k(t-1)$$
$$\Delta b_i^k(t) = \eta_b \rho_i^k(t) + \alpha_b \Delta b_i^k(t-1) \quad (2)$$

Where the subscripts w and b represent the weight and threshold respectively, α_w and α_b are momentum constants which determine the influence of the past parameter changes on the current direction of movement in the parameter space, η_w and η_b represent the learning rates, also, $\rho_i^k(t)$ is the error signal of the i-th neuron of the k-th layer which is back propagated into the network. Since the activation function of the output neuron is linear, the error signal at the output node is

$$\rho^m(t) = y(t) - \hat{y}(t) \tag{3}$$

and for the neurons in the hidden layer

$$\rho_i^k(t) = F'\left(v_i^k(t)\right)\sum_j \rho_j^{k+1}(t) w_{ji}^{k+1}(t-1)$$

$$k = m\text{-}1, \ldots, 2, 1 \tag{4}$$

where $F'\left(v_i^k(t)\right)$ is the first derivative of

$F\left(v_i^k(t)\right)$ with respect to $v_i^k(t)$.

Back Propagation algorithm suffers from a slow convergence rate which verifies algorithm's reputation of being steepest. The search for the global minima may become trapped at local minima and the algorithm can be sensitive to the user selectable parameters.

III) DELTA RULE

The Delta Rule is a further variation of Hebb's Rule , and it is one of the most commonly applied ways available to modify the strengths of the input connections in order to reduce the difference between the desired output value and the actual output of neuron . This rule changes the connection weights in the way that minimizes the mean squared error of the network . The error is back propagated into previous layers one layer at a time . The process of back-propagating the network errors continues until the first layer is reached . The network type called Feed forward , Back-propagation derives its name from this method of computing the error team . This rule is also referred to as the Windrow-Hoff Learning Rule and the Least Mean Square Learning Rule .

IV) PERCEPTRON

The perceptron is the simplest form of a neural network used for the classification of a special type of patterns, which are linearly separable. It consists of a single McCulloch-Pitts neuron with adjustable synaptic weights and bias (threshold) [12] ,proved that if the patterns (vectors) used to train the perceptron are drawn from linearly separable classes, then the perceptron algorithm converges and positions the decision surface in the form of a hyperplane between the classes. The proof of convergence of the algorithm is known as the perceptron convergence theorem.

The single-layer perceptron shown has a single neuron. Such a perceptron is limited to performing pattern classification with only two classes.

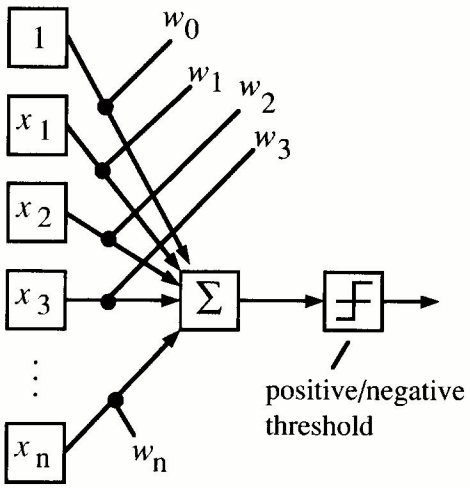

Figure 1. Perceptron.

$$v(x) = \sum_{i=0}^{n} w_i x_i = w^T x \tag{5}$$

$$y = \begin{cases} 1, & v \geq 0 \\ 0, & v < 0 \end{cases} \tag{6}$$

Equation v(x) = 0 defines a boundary between the region where the perceptron fires at, and the region where it outputs zero. This boundary is a line (decision line, decision hyperplane), which must be appropriately located during the process of learning. The perceptron can distinguish between empty and full patterns if and only if they are linearly separable.

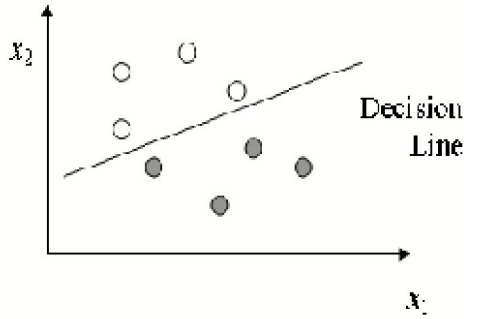

Figure 2. Linearly separable sets.

Training set T is a set of pairs $[x^i, d^i]$, where the desired value di equals either 1 or 0 and

$$T = \begin{bmatrix} [x^1 & d^1] \\ [x^2 & d^2] \\ [x^N & d^N] \end{bmatrix}$$

$$x^i = \begin{bmatrix} x_0^i & x_1^i & \cdots & x_n^i \end{bmatrix}^T \tag{7}$$

The training instance (sample) $[x^m, d^m]$ is misclassified if the perceptron output y^m does not produce $d^m.A^n$ appropriate measure of misclassification is this criterion:

$$J = \sum_{i=1}^{N} (y^i - d^i) v(x^i) = \sum_{i=1}^{N} (y^i - d^i) w^T x^i \quad (8)$$

Let us discuss individual terms in the criterion J.

1. If the training sample is correctly classified then $(y^i - d^i) = 0$.
2. If $d^m = 1$ and $y^m = 0$ then $(y^m - d^m) = -1$ and $(y^m - d^m) v(x^m) > 0$. (It follows from (6) that the output $y^m = 0$ only if $v(x^m) < 0$.)
3. If $dm = 0$ and $y^m = 1$ then $(y^m - d^m) = 1$ and $(y^m - d^m) v(x^m) \geq 0$. (It follows from (6) that the output $y^m = 1$ only if $v(x^m)$ 0.)

We can mine from the above, that the criterion J grows if training samples are misclassified and J = 0 if all samples are classified correctly.

A gradient descent method will be used to minimize the criterion J. Let w(k) be the k-th iteration of the weight vector w. The gradient descent method is based on the formula

$$w(k+1) = w(k) - \eta grad(J(w(k)) \quad (9)$$

$$grad(J) = \frac{\partial J}{\partial w} = \sum_{i=1}^{N} (y^i - d^i) x^i \quad (10)$$

The learning coefficient controls the size of a step against the direction of the gradient (because of a minus sign). If is too small learning is slow; if too large the process of the criterion minimization can be oscillatory. The optimum value of the learning coefficient is usually found experimentally. If is kept constant we speak about a fixed-increment learning algorithm.

V) EXPERIMENT DESIGN

In this experiment we are testing the Back Propagation, Delta Rule and Perceptron, to find out the best algorithm of learning in order to train our data. This will be achieved by providing the neural network structure by the learning algorithm and the training samples to learn. Our sample consists of distinct 12 figures for fish, seven used for trained neural network and five used for tested. The following table shows our test parameters used in the neural network.

Table 1: Test parameters used in the neural network

Number of Samples	Number of Inputs (Total)	Learning Rate	Activation Function	Learning Error Limit	Momentum
7	166	0.1	Sigmoid Function	0.001	1

VI) RESULTS AND DISCUSSIONS

A successful completion of test run for Back Propagation algorithm based on the parameters show in table 1 above, following results are observed:

Table 2. Back Propagation Results

Training Algorithm	NO. Neurons	NO. Iterations	Elapsed Time	Actual Error
Back Propagation	3	1277	15 sec	3.55186

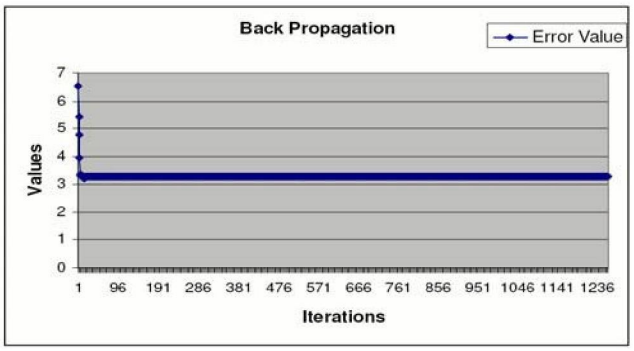

Figure 3. Back Propagation Training

The experiment results reveal that back propagation algorithm has failed as each of the seven samples has an output with 7 digits. Even if we add up the total number of input which is 166 with the output of 35, the total will be 101 as input, which is not even close to the expected output value for the learning rule of back propagation.

We run the test for delta rule algorithm based on the parameters show in table 1, and we get the following results as show in table 3.

Table 3: Delta rule Results

Training Algorithm	NO. Neurons	NO. Iterations	Elapsed Time	Actual Error
Delta Rule	-	2866	30 sec	4.9999

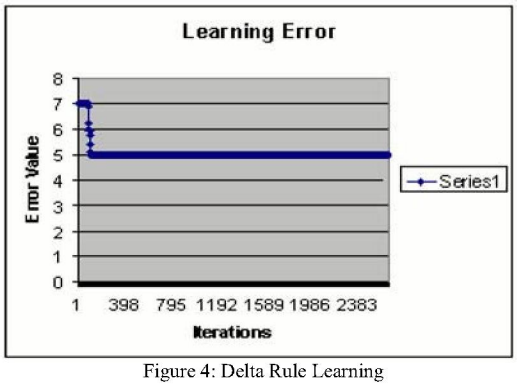

Figure 4: Delta Rule Learning

The algorithm, Delta Rule Learning, has been examined with more than 2500 iterations, and 30 seconds, and it has not given any promising results that can be used to train our data.
Finally we test the perceptron algorithm and table 4 shows our results.

Table 4. perceptron learning result

Training Algorithm	NO. Neurons	NO. Iterations	Elapsed Time	Actual Error
Perceptron		25958	45 minute	0.0019999

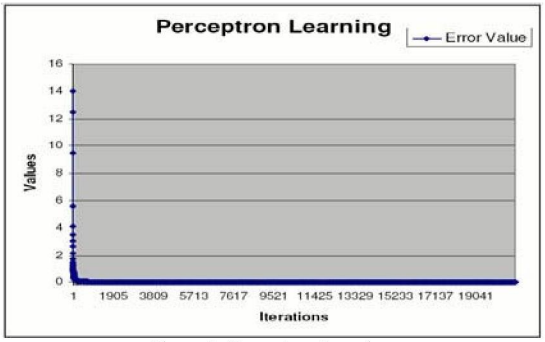

Figure 5. Perceptron Learning

As it is apparent from table 4 and figure 5, that Perceptron teacher has been tested with seven samples which were composed of total 191 inputs to the network. The algorithm has been tested with more than 20,000 iterations within a spam of 45 minutes. Graph reveals that the first value of the error was around 14 which is very prompt change to some value around 0 and then it jumps to the desired error value, the below figure 6 illustrate the comparisons between MLP algorithms .

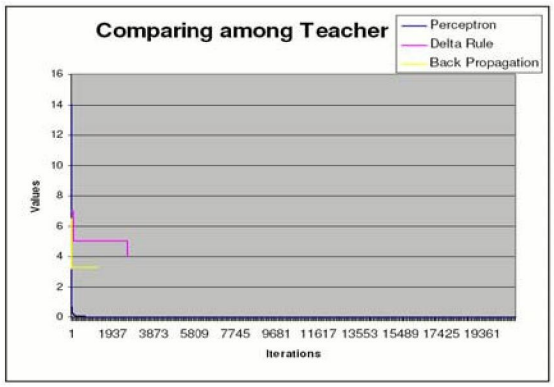

Figure 6. Comparisons between three algorithms

The reason of the stopping of Delta Learning and Back Propagation Learning is that they stopped before the Perceptron which worked for a longer time than the other two algorithms, hence making it obvious that both Delta Rule and Back Propagation have proved that they cannot be improved any further.

Perceptron teacher has proved itself as the best comparing to the other two which were examined previously. One of the tasks of Perceptron is classification [3], and for that reason Perceptron justifies the fact that it has been more successful than the rest.

VII) CONCLUSION

In short, this paper demonstrated the study of multi-layer perceptron (Back Propagation, Delta Rule and Perceptron) algorithms in neural networks and explained each algorithm. Eventually experimental findings revealed that the perceptron algorithm is the best algorithm to be used in the multi-layer perceptron in a neural network.

REFERENCES

[1] Anderson J A .An Introduction to Neural Networks.MIT Press, Cambridge, MA ,1995.

[2] Chauvin Y, Rumelhart D E (eds.). Backpropagation: Theory, Architectures, and Applications. Erlbaum, Mahwah, NJ,1995.

[3] Freund, Y. and Schapire, R. E. Large margin classification using the perceptron algorithm. In Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT' 98). ACM Press,1998.

[4] Kanal L N .On pattern, categories, and alternate realities. KS Fu award talk at IAPR, The Hague. Pattern Recognition Letters 14: 241-55, 1992.

[5] L. Gupta and M.D. Srinath. Contour sequence moments for the classification of closed planar shapes .Pattern Recognition. Vol. 20-3. 1987.

[6] L. Gupta, M.R Sayeh. and R. Tammana. A neural network approach to robust shape classification. Pattern Recognition vol. 23-6, 1990.

[7] Parisi. D. Artificial Life and Higher Level Cognition, Brain and Cognition, Volume 34, Issue 1, June 1997, Pages 160-184, 2002.

[8] Qiyao Yu, C. Moloney and F. M. Williams. SAR Seaice Texture Classification using Discrete Wavelet Transform Based Methods .IEEE International Geoscience and Remote Sensing Symposium, vol. 5, pp. 3041-3043, 2002.

[9] Rumelhart, D.E., and McClelland, J.L. *Parallel distributed processing: explorations in the microstructure of cognition. I & II*, MIT Press, Cambridge, MA,1986.

[10] Veerendra Singh and S. Mohan Rao. Application of image processing and radial basis neural network techniques for ore sorting and ore classification .Minerals Engineering, vol. 18, pp. 1412-1420, Dec. 2005.

[11] Werbos, P.J. Beyond Regression: New Tools for Prediction and Analysis in the Behavioural Sciences. Ph.D. Thesis, Harvard University,1974.

[12] Philip M. Merikle D ,Smilek ,John D.Perception without awareness: perspectives from cognitive psychology. Sscience direct, Volume 79, Issues 1-2, April 2001, Pages 115-134, 2001.