Multilayer Perceptron Network in HIV/AIDS Application

Manaswini Pradhan[1] , Ranjit Kumar Sahu[2]

[1]*P.G. Department of Information and Communication Technology, Fakir Mohan University, Orissa, India*

[2]*Consultant, Plastic ,Cosmetic and Laser Surgery, Mumbai, India*

[1]ms.manaswini.pradhan@gmail.com, [2]drsahurk@yahoo.co.in

***Abstract*--The paper discusses a new method of Multi Layered Perceptron (MLP) network to classify HIV/AIDS infected and non-infected status of individuals. For this purpose, seven features on the basis of patient's unique like age, sex, weight, HB, CD4, CD8 and TB were used as input data. In order to determine the applicability and best performance of the MLP network, three different training algorithms like Back Propagation, Levenberg-Marquardt, and Bayesian Rule algorithms, were employed to train the MLP networks. The findings conclude that the MLP network trained using Back Propagation algorithm produced the best performance with 89.80% accuracy as compared to Levenberg-Marquardt and Bayesian Rule algorithms. The results also significantly demonstrated the suitability of the MLP network for calculating and specifying the HIV/AIDs positive/negative status of the patient.**

***Keywords*--Multi Layered Perceptron (MLP), Back propagation, Levenberg-Marquardt, Bayesian, Regimens, HIV/AIDs.**

I. INTRODUCTION

Acquired Immunodeficiency Syndrome (AIDS) was first defined in 1982 to describe the cases of unusual immune system failure due to an unknown and unidentified infection in the previous years. The Human Immunodeficiency Virus (HIV) was later identified as the cause of AIDS. In a large number of medical applications, classification is desired to differentiate a pattern of low frequency from a pattern of high frequency. Clinical trail system for HIV/AIDs is a complex one. It is an incurable disease. More than millions of people are HIV positive. Recent research shows that computational intelligence has been widely used on medical diagnosis to solve complex problems by developing decision support system with the application of Neural Network algorithms. Neural Network is an appropirate application to practice most of the medical problems. It has many algorithms for classification, prediction, image processing, etc. A proper utilization of a Neural Network technique to implement a large-scale health services research dataset is one of the most difficult areas in the Neural Network field. Due to incorrectly defined and unstructured factors, it becomes further complicated affecting the functional health status of HIV/AIDS patients. Many of the studies have applied Neural Network technique to classify and predict desired solution or to improve methodological aspects.

Multilayered perceptron(MLP) network trained using back propagation(BP) algorithm is one of the most popular choice in neural network applications. The present study proposes the MLP network to classify the HIV positive/negative individuals and compare the performance of various available training algorithms namely back propagation, Levenberg-Marquardt and Bayesian rule.

II. REVIEW OF RELATED RESEARCH

The Human Immunodeficiency Virus (HIV) is one of the main causes of human death in the world. The HIV is a human pathogen that infects certain types of lymphocytes called T-helper cells, which are important to the immune system. Without a sufficient number of T-helper cells, the immune system is unable to defend the body against infections, thereby making it vulnerable to various infections and diseases and finally it succumbs.

Acquired immunodeficiency syndrome (AIDS) was first defined [1] in 1982 to describe the first cases of unusual immune system failure that were identified in the previous year.

The human immunodeficiency virus (HIV) was later identified as the cause of AIDS.As an indicator, the risk factor epidemiology examines the individual demographic and social characteristics and attempts to determine factors that place an individual at risk of acquiring a life-threatening disease [2]. The demographic and social characteristics of the individuals and their behavior are used to determine the risk of HIV infection; referred to as biomedical individualism [2], [3]. By identifying the individual risk factors that lead to the HIV infection, it is possible to modify social conditions, which give rise to the disease, and thus design effective HIV prevention policies. A model will be created and used to classify the HIV status of individuals based on demographic properties. In this study, the model is created using autoencoder neural networks and genetic algorithms, which have been applied for classification.

An artificial neural network (ANN) is an inter-connected structure of processing elements. The ANN structure [4] used for this study consists of three main components (Fig. 1) [5]. These are the input layer, the hidden layer and the output layer.

Neural networks have been successfully used for medical informatics, for decision making, clinical diagnosis, prognosis, and prediction of outcomes [6]-[10] and also for classification. Marwala [11] used a probabilistic committee of neural networks to classify faults in a population of nominally identical cylindrical shells and obtained an accuracy of 95%, in classifying eight classes of fault cases. Ohno-Machado [12] depicted the limitation on the accuracy of the neural network model due to lack of data balance and increased the accuracy by using sequential neural networks. Lisboa [13] assessed the evidence of healthcare benefits using neural networks. Fernandez and Caballero [14] used ANN to model the activity of cyclic urea HIV-1 protease inhibitors. They showed that ANN was capable of representing the nonlinearity in the HIV model. Lee and Park [15] applied neural networks to classify and predict the symptomatic status of HIV/AIDS patients based on publicly available HIV/AIDS data. A study was also performed to predict the functional health status of HIV/AIDS patients defined as 'in good health' or 'not in good health', using neural networks [16]. Laumann and Youm [17] used the racial and ethnic group differences to model the prevalence of the disease and succeeded in relating the demographic properties to the transmission of the disease. Poundstone and others [2] related demographic properties to the spread of HIV. Their work justified the use of such demographic properties in creating a model to predict the HIV status of individuals, as done in the present paper.

All the models refereed above concluded that ANN performs better in HIV classification problems. The methodology adopted here aims at using demographic and social factors, to predict the HIV status of an individual, using autoencoder neural networks. The most common neural network architecture is the multilayer perceptron (MLP). An alternative network is the Radial Basis Function (RBF) [5]. The use of MLP over RBF can be attributed to the fact that the RBF usually requires the implementation of the pseudo-inverse of a matrix for training, which is often singular while MLP uses conventional feedforward optimization methods, which are stable[5]. In the present case, preliminary design showed that the MLP has outperformed. This can be attributed to the fact that MLP networks, also known as universal approximators, are capable of modeling any complex relationship with one or two hidden layers and are thus most suited for this study. For a detailed analysis on neural networks and MLP one can refer to the studies made by [18]-[22] For the purpose of the present paper, neural networks are used with genetic algorithms. A genetic algorithm (GA) is an optimization method deriving its behavior from processes of evolution in nature, inspired by Darwin's theory of natural evolution [23],[24]. This is done by the creation of a population taking individuals within a machine/computer. In this study, the population of individuals represents the missing input entries. The individuals then go through the process of evolution. GA uses fitness-proportionate or tournament selection to select the missing entries (individuals), probabilistically that yields the right HIV status for the individuals. Although not guaranteed to provide the globally optimum solution, GA has been shown to be highly efficient at reaching to a very near optimum solution in a computationally efficient manner [23],[24]. For more details on GA one can refer to Davis and Michalewicz [25],[26]. In the literature review, there is no method proposed thus far that investigates the use of Autoencoder networks for HIV modeling which is based on autoassociative models [27] combined with GA to classify the HIV status of an individual based on demographic properties.

Multilayered perceptron (MLP) network trained using back propagation (BP) algorithm is the most popular choice in neural network applications. The present study proposes the MLP network to predict HIV/AIDS Regimen specification and compare the performance of various available training algorithms namely back propagation, Levenberg-Marquardt and Bayesian rule.

This paper is sequenced as follows. Section 3 discusses the basic concept of the MLP network. Section 4 outlines the training algorithms employed in this research. Section 5 discusses the methodology. Section 6 reflects the result and discussion. Finally, section 7 outlines the conclusion.

III. MULTILAYERED PERCEPTRON NETWORK

A MLP network is a feed forward neural network with one or more hidden layers. Cybenko and Funahashi [28], [29] have proved that the MLP network is a general function approximator and the MLP network with one hidden layer (as shown in Fig. 1) is sufficient to approximate any continuous function. Based on Fig. 1, the input layer acts as an input data buffer that distributes the input to the hidden layer. The outputs from the hidden layer then become the inputs to the output layer, which provides the network output.

A hidden neuron performs two functions, i.e. the combining function and the activation function. Consider a MLP network with n_i input nodes, the output of the j-th neuron of the hidden layer is given by:

$$v_j(t)=F\left(\sum_{i=1}^{n_i} w_{ji}^1 x_i(t)+b_j\right);\ for 1 <= j <= n_h \qquad (1)$$

where the w_{ji}^1 denotes the weights that connect the input and hidden layer; x_i and b_i denote the input that are supplied to the input layer and thresholds in hidden nodes respectively; n_i *and* n_h are number of input nodes and hidden nodes respectively.

The output of the *k*-th output neuron, y_k in the output layer is given by:

$$\hat{y_k}(t) = \sum_{j=1}^{n_h} w_{kj}^2\ v_j(t); \text{for } 1<=k<= n_o \qquad (2)$$

where n_o is the number of output nodes; w_{kj}^2 denotes the weights of the connections between the hidden and output layer. It can be derived from equations (1) and (2) that the MLP network with one hidden layer can be expressed by following equation:

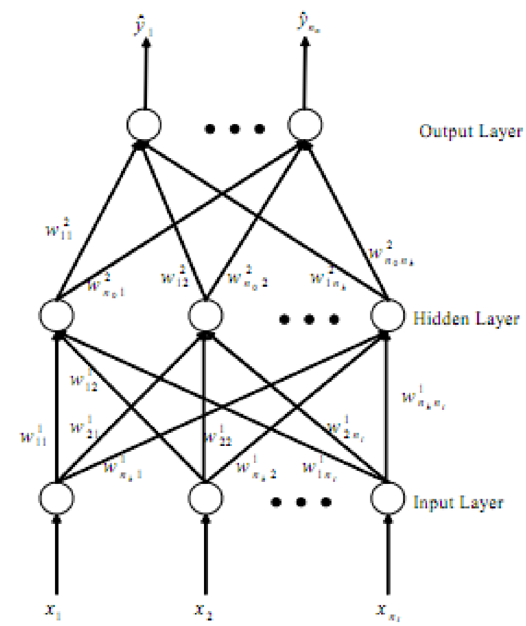


Fig. 1 Multilayered Perceptron Networks

$$\hat{y_k}(t)=\sum_{j=1}^{n_h} w_{kj}^2\ F\left(\sum_{i=1}^{n_i} w_{ji}^1\ x_i(t) + b_j\right);\ \text{for } 1 \leq k \leq n_o \qquad (3)$$

F(.) is an activation function that is normally selected as a sigmoid function, which is given by:

$$F(v(t)) = \frac{1}{1+e^{-v(t)}} \qquad (4)$$

The weights $w_{ji}^1,\ w_{kj}^2$ and threshold b_j are unknown and should be selected to minimize the prediction errors, defined as:

$$\rho_k(t)=y_k(t) - (t) \qquad (5)$$

where $y_k(t)$ is the actual output and $\hat{y_k}(t)$ is the network output.

IV. TRAINING ALGORITHM

As described earlier, three training algorithms for the MLP network will be employed and compared for their performance. This section briefly presents the back propagation, Levenberg-Marquardt and Bayesian Rule algorithms.

A. *Back Propagation Algorithm*

Back-propagation, the most commonly used to train The MLP network, is a gradient descent procedure that computes the derivatives' values in a very efficient way, and modifies the weights according to a parameter known as learning rate [30]. Back propagation is a steepest decent type algorithm where the weight connection between the *j*-th neuron of the hidden layer and the *i*-th neuron of the input layer are respectively updated according to:

$$\begin{aligned} w_{ji}(t) &= w_{ji}(t-1) + \Delta w_{ji}(t) \\ b_j(t) &= b_j(t-1) + \Delta b_j(t) \end{aligned} \qquad (6)$$

with the increment $\Delta\ w_{ji}(t)$ and $\Delta\ b_j(t)$

given by:

$$\begin{aligned} \Delta w_{ij}(t) &= \eta_w \rho_j(t) x_i(t) + \alpha_w \Delta w_{ji}(t-1) \\ \Delta b_j(t) &= \eta_b \rho_j(t) + \alpha_b \Delta b_j(t-1) \end{aligned} \qquad (7)$$

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 and $\rho_j(t)$ is the error signal of the *j*-th neuron of the hidden layer which is back propagated in the network. Since the activation function of the output neuron is linear, the error signal at the output node is

$$\rho(t) = y_k(t)\hat{y}_k(t) \quad (8)$$

and for the neurons in the hidden layer

$$\rho_j(t) = F^{'}(x_i(t))\sum_j \rho_j^k(t) w_{ji}^2(t-1) \quad (9)$$

where $F^{'}(x_i(t))$ is the first derivative of $F(x_i(t))$ with respect to $x_i(t)$.

Since back propagation algorithm is a steepest decent type algorithm, the algorithm suffers from a slow convergence rate. The search for the global minima may be trapped at local minima and the algorithm can be sensitive to the user selectable parameters (Mashor, M. Y. 2003)

B. Levenberg-Marquardt Algorithm

Levenberg-Marquardt algorithm is a gradient-based, deterministic local optimization algorithm. The LevenbergMarquardt algorithm has an advantage over the traditional Back Propagation algorithm, where it can provide faster (second-order) convergence rate and keep relative stability. [32],[33].

Like the quasi-Newton methods, the Levenberg-Marquardt algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training feed forward networks), then the Hessian matrix can be approximated as:

$$H = J^T J \quad (10)$$

and the gradient can be computed as:

$$g = J^T e \quad (11)$$

where J is the Jacobian matrix that contains first derivatives of the network errors with respect to the weights and biases, and *e* is a vector of network errors. The Jacobian matrix can be computed through a standard back propagation technique that is much less complex than computing the Hessian matrix [34].The Levenberg-Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update:

$$\Delta w = -\left[J^T J + \mu I\right]^{-1} J^T e \quad (12)$$

where Δw is a differential weights and μ is a control parameter.

When the scalar μ is zero, it is similar to Newton's method, using the approximate Hessian matrix. When μ is large, it becomes gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so the aim is to shift towards Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase the performance function. In this way, the performance function will always be reduced at each iteration of the algorithm [34].

C. Bayesian Rule Algorithm

Given the Baye's Rule as

$$P(\theta \mid D) = \frac{P(D \mid \theta)}{P(D)} \quad (13)$$

Where $P(\theta)$ is the prior probability of a parameter θ before having seen the data and $p(\theta \mid D)$ called the likelihood were the probability of the data D.

Bayes' Rule is used to determine the posterior probability of θ given the data D [34]. In general this will provide an entire distribution over possible values of θ. This process was applied to neural networks and come up with the probability distribution over the network weights, w, given the training data. When finding a posterior distribution over weights,

$$p(w \mid D) = \frac{p(D \mid w)\,p(w)}{p(D)} = \frac{p(D \mid w)}{\int p(D \mid w)\,p(w)\,dw} \quad (14)$$

In the Bayesian formalism, learning the weights means changing our belief about the weights from the prior, $p(w)$, to the posterior $p(w \mid D)$, as a consequence of seeing the data as illustrated by Fig. 2.

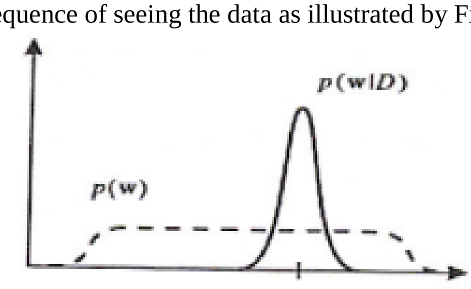


Fig. 2 Changing prior weights to posterior weights

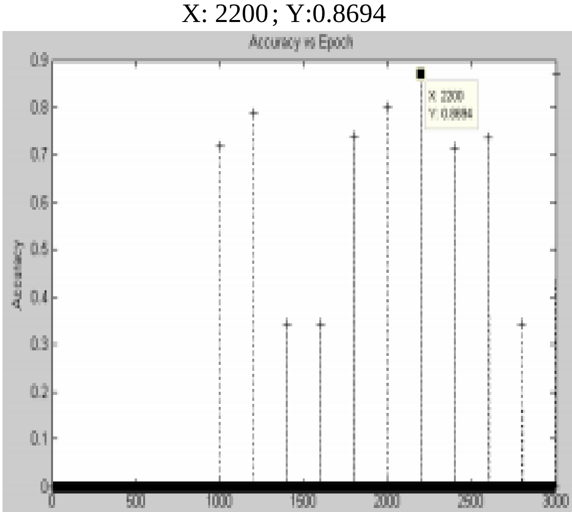
V. Methodology And Data Samples

As discussed earlier, the study focuses on classification of HIV/AIDS infected and non-infected individuals. To determine the applicability of the MLP network as HIV/AIDS diagnosis technique, the MLP network needs to go through training and testing phases. During both phases, the optimum structure and diagnosis performance of the MLP networks are determined. The performance analysis of the MLP network is based on accuracy. Accuracy is defined as the percentage of overall correct determination of HIV/AIDS cases. The data are taken from National AIDS Control Organization (NACO), India. Seven unique factors like age, sex, weight, HB, CD4, CD8 and TB used as input data to the MLP network. 200 Patient's medical information was used as training data while the remaining 100 data were used as testing data. The data are fed randomly into the MLP network. It was implemented using MATLAB 7.10 neural network tool (nntool). An input layer is used to represent set of input variables (seven input variables). Input pattern has the seven variables: age, sex, weight, CD4 count, CD8 count, HB rate and TB are taken as network parameter.

VI. Result And Discussion

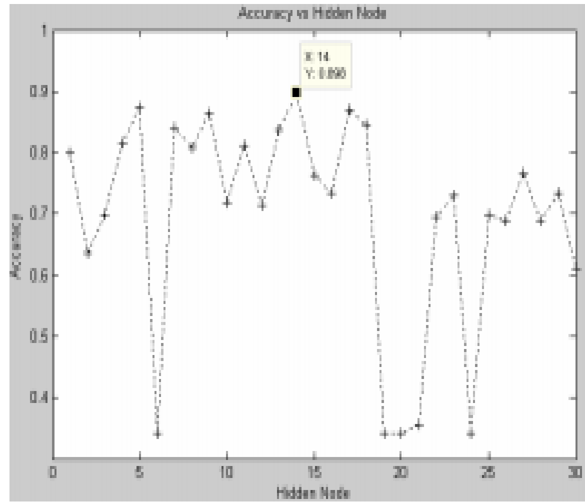
By performing the relevant training, the optimum numbers of hidden nodes and training epochs were obtained. This was obtained when the MLP network achieves the highest performance. Fig. 3 (a) and (b) show the result of obtaining the optimum training epochs and number of hidden node for back propagation algorithm respectively. The MLP network using back propagation algorithm achieves the highest performance at the number of hidden nodes equal to 2200 and 14 for training epochs. This records the optimum level and is also the highest using the parameters referred earlier.

X: 2200; Y:0.8694



(a) Training Epochs

X: 14 ; Y: 0.898

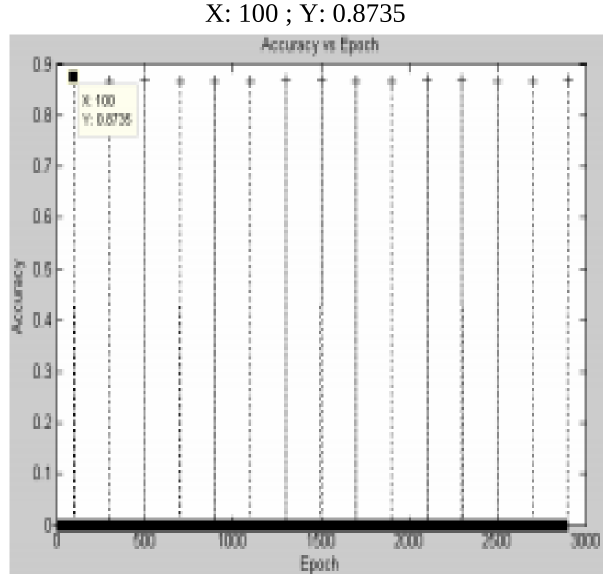


(b) Hidden Node

Fig. 3 Performance of the MLP network with Back propagation algorithm

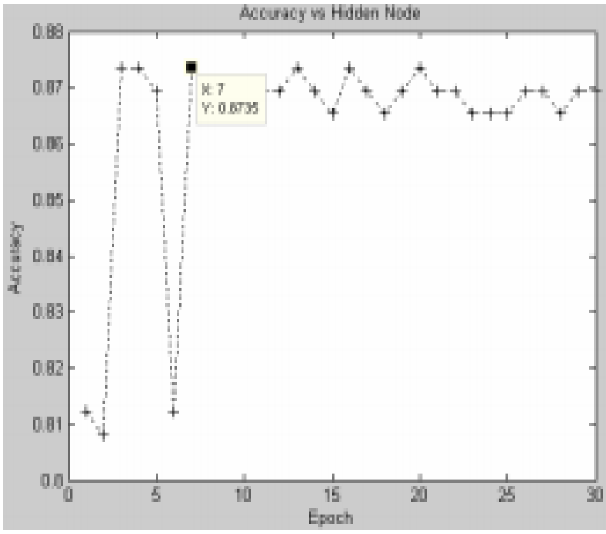
Results for Levenberg-Marquardt training algorithm are shown in Fig. 4.This training algorithm produced the best performance at 100 training epochs and 7 hidden nodes

X: 100 ; Y: 0.8735



(a)Training Epoches

X: 7 ; Y: 0.8735



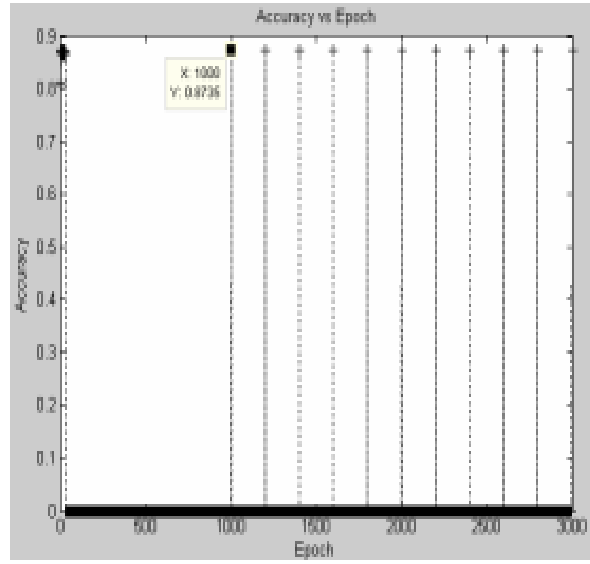
(b)Hidden Node

Fig. 4 Performance of the MLP network with Levenberg-Marquardt algorithm

Fig. 5 shows the result for the Bayesian Rule training algorithm. This training algorithm achieved an optimal result at 1000 training epochs and 3 hidden nodes.

After obtaining the optimum structure for the network, the performance of the MLP network was determined. The following Table 1 shows the performance comparison of the MLP network using the three training algorithms. A comparative analysis of the results of the three training algorithms shows that back propagation training algorithm produces the highest accuracy, 89.80% as compared to Levenberg-Marquardt and Bayesian Rule training algorithm, which produces 87.35% and 87.76% of accuracy respectively. This highest accuracy is obtained keeping all the other factors constant for the three training algorithms.

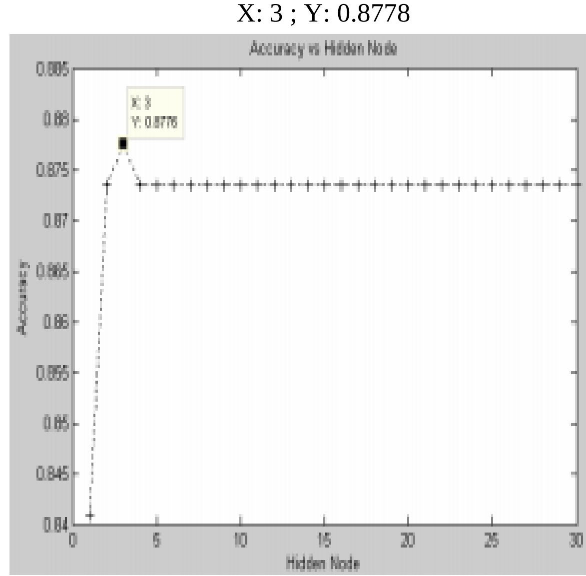
X: 1000 ; Y: 0.8735



(a) Training Epoches

(b)

X: 3 ; Y: 0.8778



(b) Hidden Node

Fig. 5 Performance of the MLP network with Bayesian Rule algorithm

TABLE I
THE PERFORMANCE COMPARISON OF THE MLP NETWORK WITH THREE DIFFERENT TRAINING ALGORITHMS

Training Algorithm	**Accuracy**
Back propagation	89.80%
Levenberg-Marquardt	87.35%
Bayesian Rule	87.76%

VII. CONCLUSION

The research undertaken has been implemented using MLP. In this particular model the patients were grouped into active and inactive based on its output. The network was trained and the weight of the hidden layer got adjusted. The final adjusted weight was the result of this model. The outcome of this research shows that the prediction model devised provides medical practitioners a convenient decision support tool that can be used to predict cases of HIV/AIDS infected patient. The findings of the research indicate that this prediction method is a promising method for identifying and treating HIV/AIDS patients. This system has the potential to improve the outcomes of health services and strengthen the accurate prediction of AIDS infected patients. The results obtained indicate that, the MLP network which has been trained with the back propagation algorithm produced the highest performance compared to the Levenberg-Marquardt and Bayesian Rule algorithms. The result also proved that the MLP network can be implemented to HIV/AIDS infected cases based on seven unique features that have been taken for the purpose of the research (i.e. age, sex, weight, HB, CD4, CD8 and TB) DELETE. There is also a scope for further study by applying the different types of neural network architectures combining with the other learning algorithms that can be done in order to find the most appropriate network for classification of HIV/AIDS positive and negative. HIV/AIDS is one of the major health challenges to the world health community. Millions of people are getting infected with this virus every day and thousands are dying every day throughout the world. This problem is now not limited to the under-developed countries but has spread to the developed countries also. The proportion of this problem is so high and pressing that the United Nations identified it as one of its eight Millennium Development Goals. Every effort at every level should be taken on priority basis to control the menace of this infection. HIV/AIDS is now not a concern only for the medical and health professionals, but for every people at all levels. In this effort the paper analyzes the MLP network using the three algorithms and concludes with concrete results that the back propagation algorithm provides with the highest accuracy rates and can be used in the context of HIV/AIDS.

REFERENCES

[1] Root-Bernstein, R., The evolving definition of AIDS. Rethinking AIDS. http://www.virusmyth.net/aids/data/rrbdef.htm: last accessed: 20–01–2011.

[2] Poundstone, K., Strathdee, S. and Celectano, D., The social epidemiology of human immunodeficiency virus/acquired immunodeficiency syndrome. Epidemiologic Rev., 2004, 26, 22–35.

[3] Fee, E. and Krieger, N., Understanding AIDS: historical interpretations and limits of biomedical individualism. Am. J. Public Health, 1993, 83, 1477–1488.

[4] Nelson, M. M. and Illingworth, W. T., A Practical Guide to Neural Nets, Addison-Wesley, New York , 1991, 3rd edn.

[5] Bishop, C. M., Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1995.

[6] Tandon, R., Adak, S. and Kaye, J. A, Neural network for longitudinal studies in Alzheimer's disease. Artif. Intell. Med., 2006, 36, 245–255.

[7] Alkan, A., Koklukaya, E. and Subasi, A., Automatic seizure detection in EGG using logistic regression and artificial neural network. J . Neurosc i. Me thods, 2005, 148, 167–176.

[8] Sawa, T. and Ohno-Machado, L., A neural network-based similarity index for clustering DNA microarray data. Comput. Biol. Med., 2003, 33, 1–15.

[9] Szpurek, D., Moszynski, R., Smolen, A. and Sajdak, S., Artificial neural network computer prediction of ovarian malignancy in women with adnexal masses. In t. J . Gynae col. Obste t., 2005, 89, 108–113.

[10] Tan, A-H. and Pan, H., Predictive neural network for gene expression data analysis. Neural Ne twork s, 2005, 18, 297–306.

[11] Marwala, T., Probabilistic fault identification using a committee of neural networks and vibration data. J . Airc raft, 2001, 38, 138–146.

[12] Ohno-Machado, L., Sequential use of neural networks for survival prediction in AIDS. Proc e edings: AMMA Annual Fa ll Symposium, 1996, pp. 170–174.

[13] Lisboa, P. J. G., A review of evidence of health benefit from artificial neural networks in medical intervention. Neural Ne twork s, 2002, 15, 11–39.

[14] Fernandez, M. and Caballero, J., Modeling of activity of cyclic urea HIV-1 protease inhibitors using regularized-artificial neural networks. J . Bioorg. Med. Ch em., 2006, 14, 280–294.

[15] Lee, C. and Park, J., Assessment of HIV/AIDS-related health performance using an artificial neural network. J . In f. Manage ., 2001, 38, 231–238.

[16] Sardari, S. and Sardari, D., Applications of artificial neural network in AIDS research and therapy. Curr. Pharmaceut. Design, 2002, 8, 659–670.

[17] Laumann, E. O. and Youm, Y., Racial/ethnic group differences in the prevalence of sexually transmitted diseases in the United States: a network explanation. Se x Transm. Dis., 1999, 26, 250–261.

[18] Hudson, D. L. and Cohen, M. E., Neural Ne twork s and Artificial Intelligence for Biomedical Engineering. IEEE Press, NJ, 2000.

[19] Deo, M. C. and Jagdale, S. S., Prediction of breaking waves with neural networks. J . Oc ean Eng., 2003, 30, 1163–1178.

[20] Narendra, K. and Lewis, F., Introduction to the special issue on neural network feedback control. Automatica, 2001, 37, 1147–1148.

[21] Rafiq, M. Y., Bugmann, G. and Easterbrook, D. J., Neural network design for engineering applications. J . Comput. Stru c t., 2001, 79, 1541–1552.

[22] Svozil, D., Kvasnicka, V. and Pospichal, J., Introduction to multilayer feed-forward neural networks. J. Chemometric s Inte ll. Lab. Sy st., 1997, 39, 43–62.

[23] Holland, J., Adaptation in Na tu ra l and Artificial Systems, University of Michigan Press, Ann Arbor, 1975.

[24] Goldberg, D. E., Gene tic Algorithms in Search Op timization and Ma chin e Learning, Addison-Wesley, Reading, 1989.

[25] Davis, L., Handbook of Gene tic Algorithms, Van Nostrand, New York , 1991.

[26] Michalewicz, Z., Genetic Algorithms + Data Structures = Evolution Programs. Berlin, Springer, 1996, 3rd edn.

[27] Lu, P. J. and Hsu, T. C., Application of autoassociative neural network on gas-path sensor data validation. J. Propul. Power, 2002, 18, 879–888.

[28] Cybenko, G. (1989), "Approximations by superposition of a sigmoidal function", Mathematics of Control, Signal and Systems, 2, 303-314.

[29] Funahashi, K. (1989). "On the approximate realisation of continuous mappings by neural networks", Neural Networks, 2, 183-192.

[30] El-Fallahi, A.A., Martí, R.A. , Lasdon, L.B. (2006). "Path relinking and GRG for artificial neural networks", European Journal of Operational Research, 169(2), 508-519.

[31] Mashor, M.Y. (2003). "Modified Recursive Prediction Error Algorithm for Training Layered Neural Network", International Journal of the Computer, the Internet and Management, 11(2), 24-36.

[32] Battiti, R. (1992). "First and secondorder methods for learning between steepest descent and Newton's method", Neural Computation, 4, 141–166.

[33] Dong Wang, Wei-Zhen Lu (2006)."Forecasting of ozone level in time series using MLP model with a novel hybrid training algorithm", Atmospheric Environment, 40(5), 913-924.

[34] Citing internet source URL:http://www.mathworks.com/access/helpdesk/help/toolbox/ nnet

[35] Bishop, C.M. (2004). Neural Network for Pattern Recognition, Oxford University Press.