International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 8, August-2014, pp: (6-11), Impact Factor: 1.252, Available online at: www.erpublications.com

Various Transfer Functions of BPN in Electricity Load Forecasting

Surinder singh[1], Er Ajay Kumar[2]
[1]M. Tech. Scholar, RPIIT Bastara, Karnal, Haryana, India
[2]Asst. Professor, Electrical Engineering, RPIIT Bastara, Karnal, Haryana, India

Abstract: Load forecasting is the technique for prediction of electrical load. In a deregulated market it is much need for a generating company to know about the market load demand for generating near to accurate power. If the generation is not sufficient to fulfil the demand, there would be problem of irregular supply and in case of excess generation the generating company will have to bear the loss. Neural network techniques have been recently suggested for short-term load forecasting by a large number of researchers. This work studies the applicability of this kind of models. The work is intended to be a basis for a real forecasting application. First, a literature survey was conducted on the subject. Most of the reported models are based on the so-called Multilayer Perceptron (MLP) network. There are numerous model suggestions, but the large variation and lack of comparisons make it difficult to directly apply proposed methods. It was concluded that a comparative study of different model types seems necessary. Back propagation in neural network is used to train neural network. Various techniques are used in BPP analysis. In this paper mean square error (MSE) is considered as performance criteria and various BPP methods are analysed on MSE criteria.

Index Terms: Neural Networks, Back Propagation.

I. INTRODUCTION

The most used thing in today's world is energy. We are using energy in various forms in our day to day life i.e. electricity, refined oils, LPG, solar energy, wind energy, chemical energies in form of batteries and many other forms. Sometimes we are extravagant and sometimes we are careful. But the aim is to provide the uninterrupted supply to the users of electricity, and to achieve the aim there must be proper evaluation of present day and future demand of power. That's why we need a technique to tell us about the demand of consumers and the exact capability to generate the power and this need LOAD FORECASTING technique.

It is used by power companies to estimate the amount of power needed to supply the demand. It tells about the scenario of present and future load demand. It has many applications including energy purchasing and generation, load switching, contract evaluation, and infrastructure development. Load forecasting has become in recent years one of the major areas of research in electrical engineering. Load forecasting is however a difficult task. First, because the load series is complex and exhibits several levels of seasonality. Second, the load at a given hour is dependent not only on the load at the previous day, but also on the load at the same hour on the previous day and previous week, and because there are many important exogenous variables that must be considered.

II. ARTIFICIAL NEURAL NETWORK

Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the network function is determined largely by the connections between elements. We can train a neural network to perform a particular function by adjusting the values of the connections (weights) between elements. Commonly neural networks are adjusted, or trained, so that a particular input leads to a specific target output. Such a situation is shown below.

There, the network is adjusted, based on a comparison of the output and the target, until the network output matches the target. Typically many such input/target pairs are used, in this supervised learning, to train a network. Batch training of a network proceeds by making weight and bias changes based on an entire set (batch) of input vectors. Incremental training changes the weights and biases of a network as needed after presentation of each individual input vector. Incremental training is sometimes referred to as "on line" or "adaptive" training.

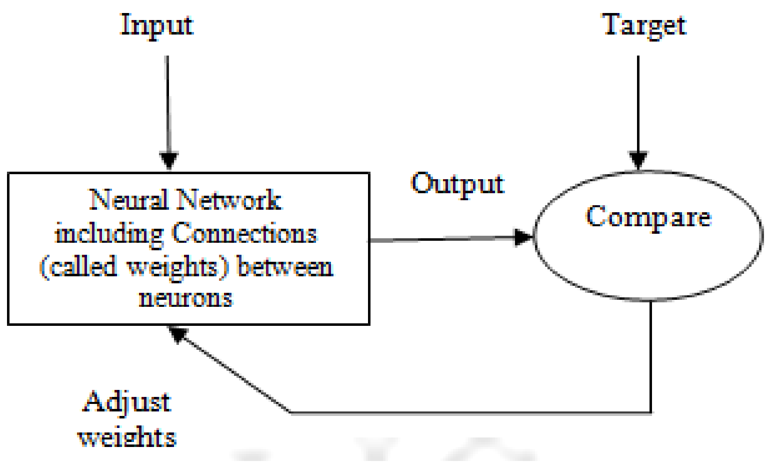

Figure 1: Flow Diagram of Neural Network Principle

Neural networks have been trained to perform complex functions in various fields of application including pattern recognition, identification, classification, speech, vision and control systems. Today neural networks can be trained to solve problems that are difficult for conventional computers or human beings. Throughout the toolbox emphasis is placed on neural network paradigms that build up to or are themselves used in engineering, financial and other practical applications. The supervised training methods are commonly used, but other networks can be obtained from unsupervised training techniques or from direct design methods. Unsupervised networks can be used, for instance, to identify groups of data. Certain kinds of linear networks and Hopfield networks are designed directly. In summary, there are a variety of kinds of design and learning techniques that enrich the choices that a user can make.

III. WORKING PRINCIPLE OF ARTIFICIAL NEURAL NETWORKS

The working principles of an artificial neural network are very straightforward. Let us take a three-layer feed forward neural network as shown in Figure 3. From left to right, starting from the input layer, each input neuron is connected to every hidden neuron in the hidden layer. Then, each hidden neuron in the hidden layer is also connected to every output neuron in the output layer. Signals are passing through the input layer and multiplied by the corresponding synaptic weights. Those multiplied signals are then summed at the hidden layer and activated by a transfer function. Let i denote the input layer, j denote the hidden layer, k denote the output layer, y_i denote an input signal, w_{ji} denote a synaptic weight between input and hidden layer, v_j denote the summed signal at a hidden neuron, and $\phi_j(.)$ denote the transfer function at the hidden layer. We could write the summing equation at the hidden layer as

$$v_j(n) = \sum_{i=1}^{n} w_{ji}(n) y_i(n) \qquad (5)$$

where n is the number of input signals. The activated sum through transfer function could be written as

$$y_j = \phi_j(v_j) + b_j \qquad (6)$$

where b_j is a threshold value at the hidden layer. The transfer function could be anyone of the step, piecewise-linear, sigmoid, or hyperbolic tangent function.

The activated signal y_j in the hidden layer would be multiplied by the synaptic weight w_{kj} between the hidden layer and the output layer and summed at the output layer. Those summed intermediate signals would then be activated by the transfer function at the output layer. Let v_k denote the summed signal at the output layer, $\phi_k(.)$ denote the transfer function, and y_k denote the output signal.

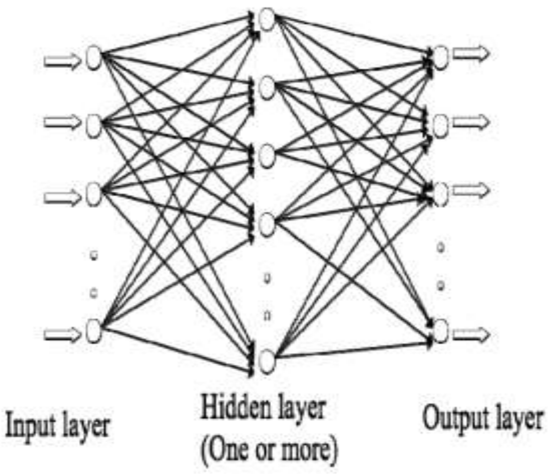

Figure 2. A Multi-Layer Fully Connected Feed forward Network

We could write the summing equation at the output layer as

$$v_k(m) = \sum_{j=1}^{m} w_{kj}(m) y_j(m) \qquad (7)$$

where m is the number of hidden neurons. The activated sum through transfer function could be written as

$$y_k = \phi_k(v_k) + b_k \qquad (8)$$

where b_k is a threshold value at the output layer. Again, the transfer function could be anyone of the step, piecewise-linear, sigmoid, or hyperbolic tangent function.
Usually a piecewise-linear function is utilized as a transfer function at the output layer.

IV. BACK PROPAGATION ARTIFICIAL NEURAL NETWORKS

The above described neural network is for forward propagation neural network. The training process of this network requires a set of examples of proper network behavior--network inputs p and target outputs t. During training the weights and biases of the network are iteratively adjusted to minimize the network performance function net.performFcn. The default performance function for feed forward networks is mean square error mse--the average squared error between the network outputs a and the target outputs t. there are various ways to train feed forward network. These use the gradient of the performance function to determine how to adjust weights to minimize performance. The gradient is determined using a technique called back propagation, which involves performing computations backward through the network. The back propagation computation is derived using the chain rule of calculus. The simplest implementation of back propagation learning updates the network weights and biases in the direction in which the performance function decreases most rapidly, the negative of the gradient. One iteration of this algorithm can be written

$$x_{k+1} = x_k - a_k g_k$$

where x_k is a vector of current weights and biases, g_k is the current gradient, and a_k is the learning rate.

The gradient descent algorithm can work in two ways: batch mode and incremental mode. In incremental mode, the gradient is computed and the weights are updated after each input is applied to the network. In batch mode, all the inputs are applied to the network before the weights are updated. The basic back propagation algorithm is 'traingd'. In this weights and biases are updated in the direction of the negative gradient of the performance function. The other algorithm is 'traingdm', gradient deviation with momentum. It allows a network to respond not only to the local gradient, but also to recent trends in the error surface. Acting like a low pass filter, momentum allows the network to ignore small features in the error surface. But these algorithms are very slow. MATLAB's neural network toolbox provides some fast algorithm which can perform hundred or thousand time faster than previously discussed. These fast algorithms can be further categorized in two categories:

1. Heuristic techniques fast algorithm
2. Standard optimization techniques fast algorithm

Back propagation algorithms under these categories are mapped as:

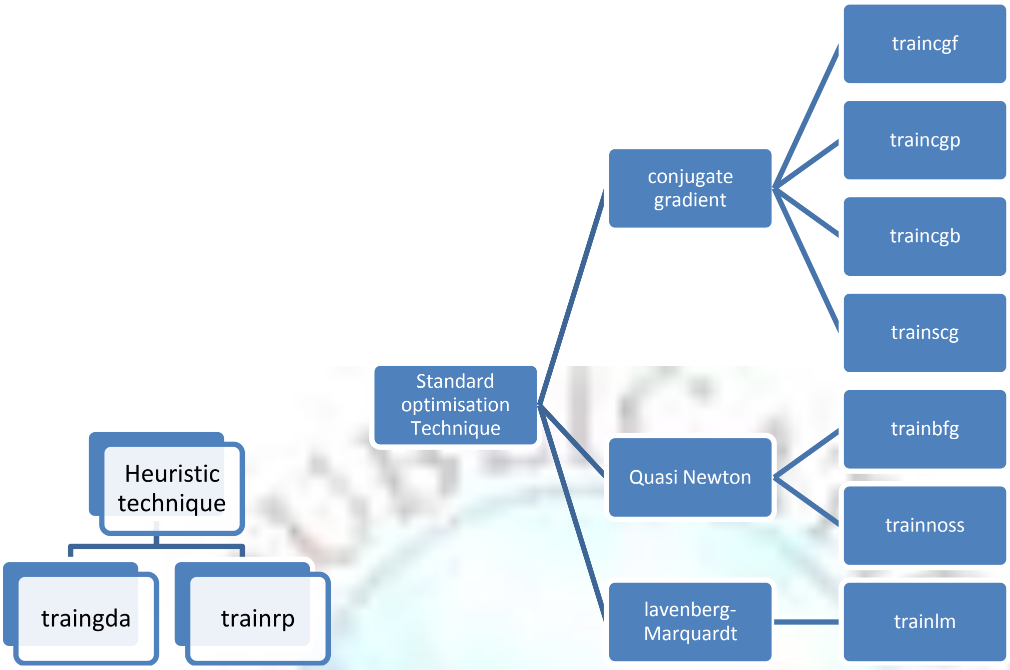

Figure 3: Chart of various back propagation algorithms

V. RESULTS

Here all type of back propagation algorithm discussed above are used as training algorithm for our network and graphs are plotted for their mean square error. These all are trained for 700 epochs and training rate = 0.03. Rest parameters are default. The trained network is retrained once again before finalizing the output. Below given graphs may differ when train again as in neural network whenever you simulate neural network you will get a different output.

For perfectly matching of trained network values with target value MSE in the network should be minimum and it should be stabilize at a value during whole simulation. Below given figures shows mean square error for two hidden layers with 14 and 24 neurons. Non linear function these hidden layers are taken ‘logsig’ and ‘tansig’.

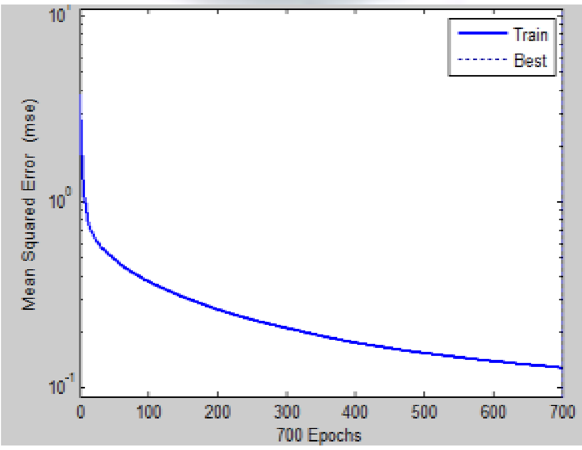

Figure 4: MSE for back propagation algorithm ‘traingd’

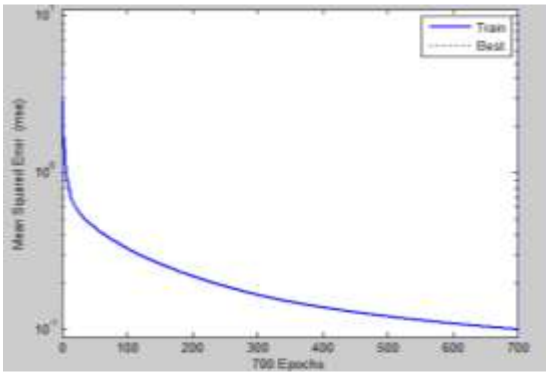

Figure 5: MSE for back propagation algorithm 'traingdm'

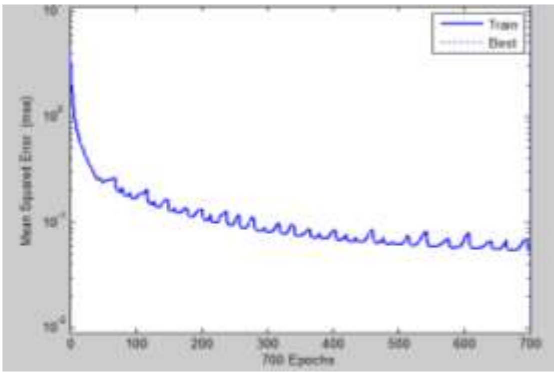

Figure 6: MSE for back propagation algorithm 'traingdma'

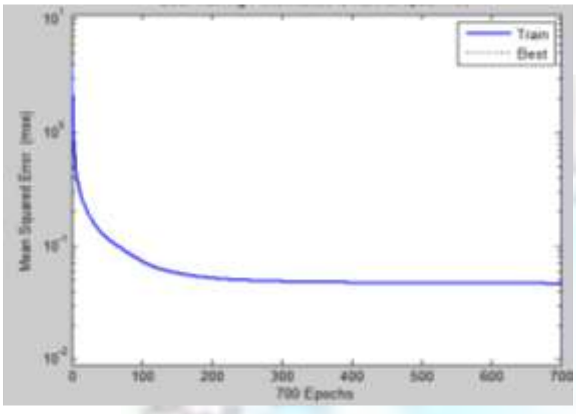

Figure 7: MSE for back propagation algorithm 'trainrp'

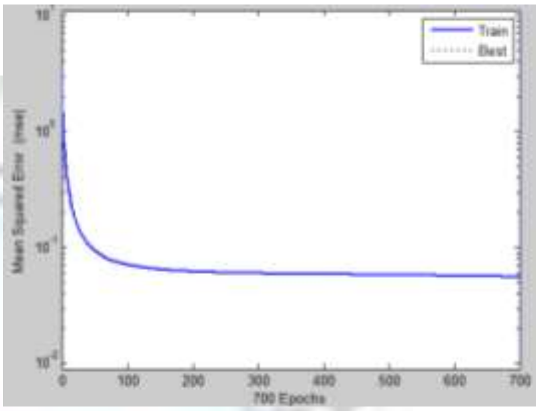

Figure 8: MSE for back propagation algorithm 'traincgf'

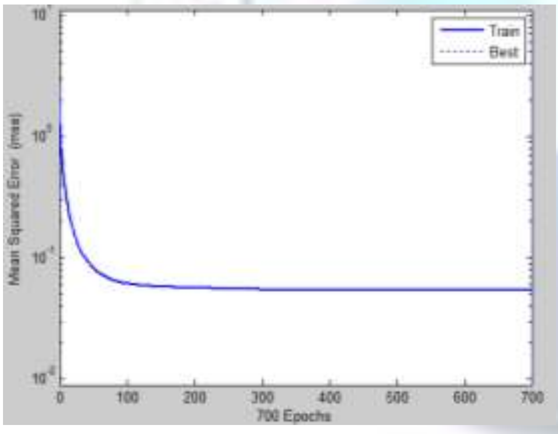

Figure 9: MSE for back propagation algorithm 'trainscg

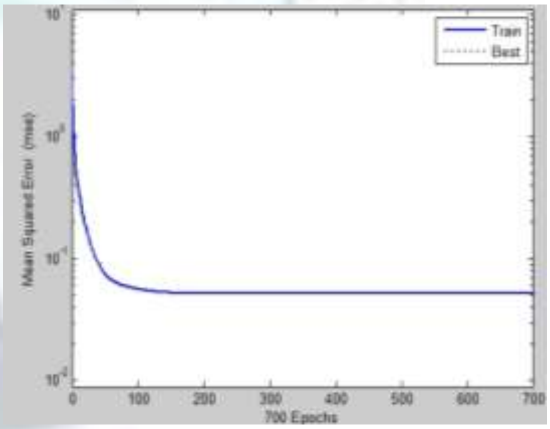

Figure 10: MSE for back propagation algorithm 'trainbfg'

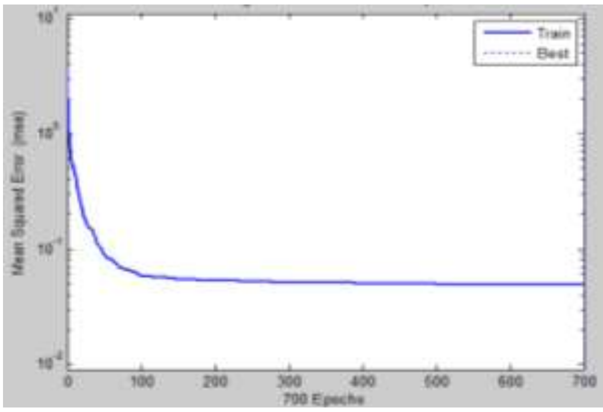

Figure 11: MSE for back propagation algorithm 'trainoss'

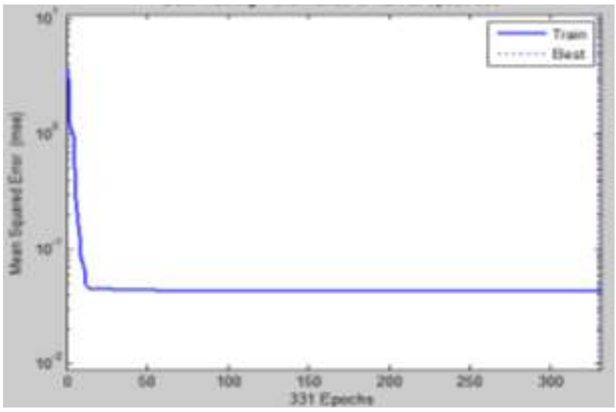

Figure 12: MSE for back propagation algorithm 'trainlm'

CONCLUSION

The result of BPP network model used for short term load forecast, shows that BPP network has a good performance and reasonable prediction accuracy was achieved for this model. Its forecasting reliabilities were evaluated by computing the mean square error between the exact and predicted values. Neural network toolbox provides a wide range of back propagation functions. Using these functions a network is tested with same number of neurons and layers. Methods based on gradient didn't give accepted results as their MSE was continually varying and amongst numerical optimization techniques 'trainlm' gives best results.

References

[1]. "Load Forecasting" Chapter 12, E.A Feinberg and Dora Genethlio, Page 269 – 285, from links: www.ams.sunysb.edu nd www.usda.gov
[2]. Moghram, S. Rahman, "Analysis and Evaluation of Five Short-Term Load Forecasting Techniques,"
[3]. Proceedings of the IEEE Transaction on Power systems, pp. 1484-1491, Vol. 4,
[4]. One Hour Ahead Load Forecasting Using Artificial Neural Network for the Western Area of Saudi Arabia by A. J. Al-Shareef, E. A. ohamed, and E. Al-Judaibi
[5]. Neural network toolbox for use with malab by Howard Demuth, Mark Beale, Martin Hogan
[6]. Artificial Neural Network Approach short Term Load Forecasting for Illam Region by Mohsen Hayati, and Yazdan Shirvany
[7]. Ho, K.-L., Y.-Y. Hsu, C-C. Yang, 1992, "short-term load forecasting using a multilayer neural network with an adaptive learning algorithm", IEEE Transactions on Power Systems, Vol. 7, No. 1, February 1992, pp. 141-148.
[8]. Gross, G., F. D. Galiana, 1987, "Short-term load forecasting", Proceedings of the IEEE, Vol. 75, No. 12, December 1987, pp. 1558-1573.
[9]. A.J. Conejo, J. Contreras, R. Esp´ınola, M.A. Plazas, "Forecasting electricity prices for a day-ahead pool-based electric energy market", Int. J. Forecast. 21 (3) (2005) 435–462.
[10]. Gupta, P. C. and K. Yamada, 1972, "Adaptive short-term load forecasting of hourly loads using weather information", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-91, No. 5, Sept./Oct. 1972, pp. 2085-2094.
[11]. Hagan, M. T. and S. M. Behr, 1987, "The time series approach to short term load forecasting", IEEE Transaction on Power Systems, Vol. PWRS-2, No. 3, August 1987, pp, 785-791.
[12]. Papalexopoulos, A. D., T. C. Hesterberg, 1990, "A regression-based approach to short-term system load forecasting", IEEE Transactions on Power Systems, Vol. 5,No. 4, November 1990, pp. 1535-1547.
[13]. Samsher Kadir Sheikh1, M. G. Unde," SHORT-TERM LOAD FORECASTING USING ANN TECHNIQUE" International Journal of Engineering Sciences & Emerging Technologies, Feb 2012 ISSN: 2231 – 6604 Volume 1, Issue 2, pp: 97-107
[14]. Salman Quaiyum, Yousuf Ibrahim Khan, Saidur Rahman, Parijat Barman," Artificial Neural Network based Short Term Load Forecasting of Power System" International Journal of Computer Applications (0975 – 8887), Volume 30– No.4, September 2011