Int. J. Adv. Res. Sci. Technol. Volume 2, Issue1, 2013, pp 30-34

IJARST

International Journal of Advanced Research in Science and Technology

journal homepage: www.ijarst.com

ISSN 2319 – 1783 (Print)

ISSN 2320 – 1126 (Online)

A Comparative study on Short term load forecasting using BPNN and Extreme Learning Machine.

J.C Pati[1], *K.K Mishra*[1], *S.K Patnaik*[2]

[1] *C. V. Raman College of Engg. BBSR, Odisha, INDIA.*
[2] *Directorate of Horticulture, BBSR, Odisha, INDIA.*

Corresponding Author's Email:* **jcpati@gmail.com

ARTICLEINFO

Article history:
Received 01 Jun. 2013
Accepted 10 Jun. 2013
Available online 10 Jun. 2013

Keywords:
Load Forecasting,
ELM,
Neural Network

ABSTRACT

Short term load forecasting plays a very important role in planning and operation of electrical utilities. Accurate forecast of electrical load is highly essential for Energy Management System. Neural network uses artificial intelligence by adjusting weights and minimizing the error. The learning speed of feed forward neural network is very slow mainly due to two reasons :- i) slow gradient-based learning algorithms to train neural networks, ii) all the parameters of the networks are tuned iteratively by using such learning algorithms. This paper presents a comparative study of back propagation algorithm and an extremely fast learning technique. A final optimise set of weight are used for prediction of actual load with greater accuracy and less time using ELM.

1. Introduction:

The short-term forecasts refer to hourly prediction of the load for a lead time ranging from one hour to several days out. Many operational decisions such as economic scheduling of the generating capacity, scheduling of fuel purchase and system security assessment are based on such forecasts. For the past several years ANNs methods received a great deal of attention and are now being proposed as powerful computational tools to solve the load forecasting problem and are able to give better performance in dealing with the nonlinearity and other difficulties in modeling time series forecasting. However due to requirement of iterative tuning of parameters and slow gradient based learning algorithm NN are slow and less accurate. Prediction accuracy and training error can be minimized by various optimization techniques such as PSO [1] and GA[2-3] etc. A new approach has been introduced in this paper using extreme learning machine. In this work weekly load for 4 months is forecasted using NN trained with back propagation technique and the results are again compared with ELM. The later technique ELM shows better result as compared to NN with high accuracy and less time.

2. Related Works:

In real applications, the neural networks are trained in finite training set. For function approximation in a finite training set, Huang and Babri[4] shows that a single-hidden layer feed forward neural network (SLFN) with at most N hidden nodes and with almost any nonlinear activation function can exactly learn N distinct observations.

Following the introduction, the remaining sections are organised as Section 2 - provides idea of neural network and BP training, Section 3- introduces ELM technique

3 Introduction to neural network:

A neural network consists mainly of three layers: input layer, hidden layer and output layer. These layer are arranged in a manner to form a multilayer feed forward network. A network model consists of simple processing elements called neurons with adjustable parameters called weights. Weighted sum of the input is passed through an activation function to produce the output. All the layers are interconnected through weights which can be optimized using training algorithm iteratively to minimize the error.

A. Model of a Neuron

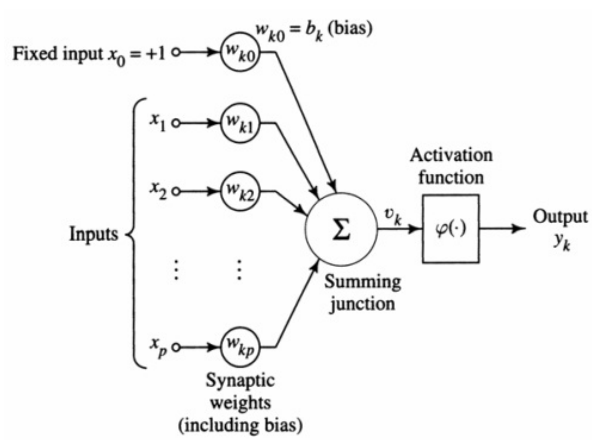

Fig.1: Model of a Neural Network

A set of synapses, each of which is characterized by a weight or strength of its own is given as input. Specifically, a signal xi at the input of synapse p connected to neuron k is multiplied by the synaptic weight wik. The weighted sum is then fed to an activation function to obtain the output.

Vk= ∑(Xi * Wik + bi) … (1)

Yk= f[Vk , ψ(.)] … (2)

The behavior of an ANN depends on both the weights and the input-output function (activation function) that is specified for the units. Typical threshold functions used are linear, threshold & sigmoid functions.

B. Backpropagation

Back Propagation is a systematic method of training multilayer artificial neural network. The performance function of the neural network is normally chosen to be the mean squared error for each pattern on the training set which is given as:-

$$E_P = 1/N\sum_{i=1}^{N}(t_{pi} - o_{pi})^2 ....(3)$$

Where t_{pi} is the target value.
o_{pi} is the output of the network.
N is the no of neurons.
The weight updates in standard BPN, is given a by:-
ΔW = W(new) – W(old) – □δE/δW
Where □ is the learning rate.

The process continues for a no of epochs or till convergence criterion is achieved and an optimized set of weight is obtained.

4. Extreme Learning Machine

Traditional algorithms used in the past decades to train neural network are quite slow and require iterative methods to obtain the output. Moreover rigorous training results to decrease in the performance or may converge to local minima. The ELM [6] method is a fast three step method for better convergence.

It has been shown that SLNFs (with N hidden neurons) and randomly chosen input weight and hidden layer bias can exactly learn N distinct observations.

A. Single layer feed forward neural networks with random hidden nodes

For N distinct arbitrary distinct samples (xj, tj) ε Rm x Rn , standard SLFNs with R hidden nodes and output function g(x) are mathematically modelled as

$\sum_{j=1}^{R} \beta\, jG(aj, bj , xk) = tk,j=1,2.....N$

where (aj , bj) are hidden node parameters.

□j is the weight vector connecting the ith hidden node and output node.

tk is the target of the kth neuron.

B. Mathematical Model

$\sum_{j=1}^{R} \beta\, jG(a_j, b_j, x_k) = t_k, j=1,2.....N$, is equivalent $H\beta = T$

$$H = \begin{pmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{pmatrix} = \begin{pmatrix} G(a_1,b_1,x_1) & \cdots & G(a_R,b_R,x_1) \\ \vdots & & \vdots \\ G(a_1,b_1,x_N) & \cdots & G(a_R,b_R,x_N) \end{pmatrix}$$

$$\beta = \begin{pmatrix} \beta_1^T \\ : \\ \beta_R^T \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} t_1^T \\ : \\ t_N^T \end{pmatrix}$$

Where H is called the hidden layer output matrix of the SLFN; the ithcoloumn of H is the output of the ith hidden node with respect to inputs $x_1, x_2 \ldots. x_n$

5. Proposed Algorithm For Computing The Hidden Layer Output Matrix USING ELM

1. Input random weights depending upon no of input and hidden neurons.

2. Input random bias matrix.

3 .Calculate H = Input weight * Input data

4. Calculate H=H + bias matrix.

5. Calculate the threshold output of the H matrix.

6. Calculate output weight as $\mathbf{H}^{+}$* target matrix.

$\mathbf{H}^{+}$ is called the Moore – Penrose generalised inverse of the hidden layer output matrix **H**.

A. Network Model for The proposed Algorithm

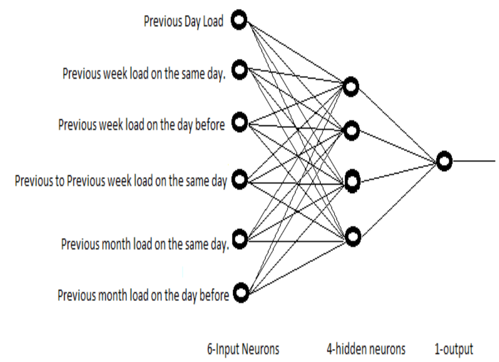

Fig.2: BPNN ELM Network Model

As per Fig:2.the neural model takes 6 inputs as previous day load, previous week load on the same day, 192hr before, previous to previous week load on the same day, previous month load on the same day, and its day before. 8 months of data are taken for training and 4 months for testing. Third week of the four months were used for forecasting. The structure of the neural network is 6-4-1.The reason for taking the input in such a way is keeping in account the daily load variation as well as seasonal variations. The training function used was trainscg , adaption learning function as learngdm, performance function as MSE and transfer function as logsig.

For prediction using ELM same 6 inputs with 4 hidden neurons were used. The threshold function used was bipolar sigmoidal function. Input weight and bias are taken in random.

5. Experimental Results

A. Datasets

For our analysis we used daily peak demand (excluding trading) spreading over April 2011 to March 2012 of Orissa Power Transmission Corporation Limited.

By using BPNN model with 4 hidden neurons the best result in MAPE was found to be 2.627% for the month of December and for the same month and with same 4 hidden layer neurons the MAPE using ELM technique was found to be 1.553%. The training accuracy in case of BPNN varied between 92.743% to 97.371%

for 25 simulations whereas in case in case of ELM the results were far better as accuracy varying from 94.098 % to 98.446%.

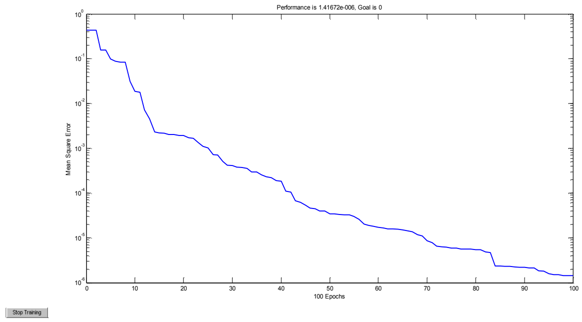

Fig.3: Convergence Plot Using BPNN

Fig: 3 shows the performance plot after training neural network using Back propagation showing mean square error and no of epochs.

B. Prediction using BPNN and ELM

A comparative study is presented using ELM and BPNN forecast model for 7 days as shown below.

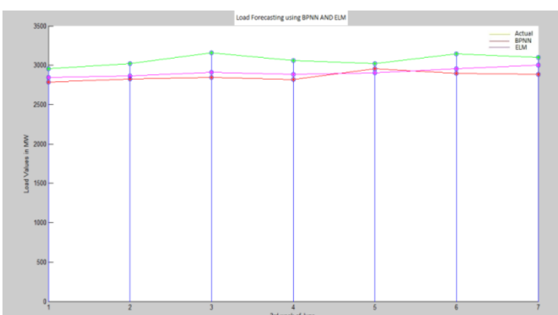

Fig4: BPNN and ELM comparison results for 3rd week of JUNE

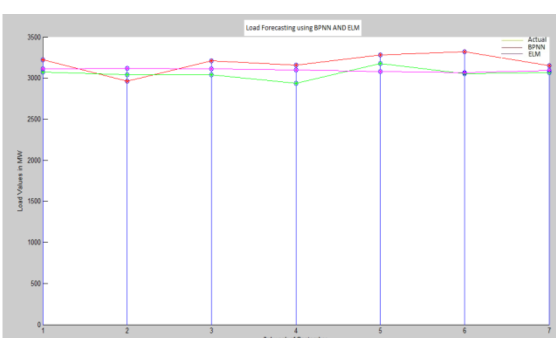

Fig5: BPNN and ELM comparison results for 3rd week of SEPTEMBER

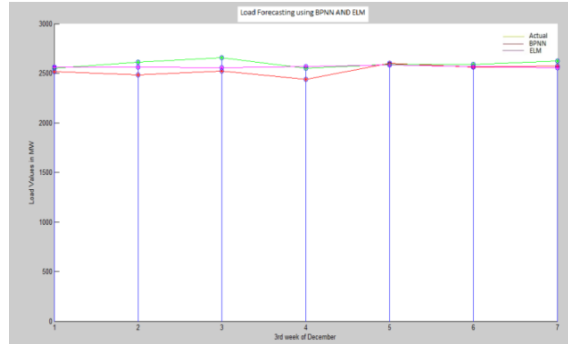

Fig6: BPNN and ELM comparison results for 3rd week of DECEMBER

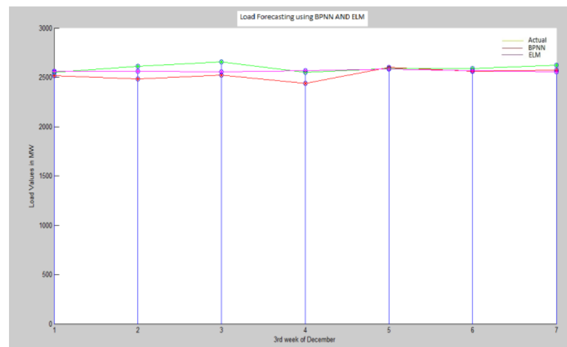

Fig7: BPNN and ELM comparison results for 3rd week of MARCH

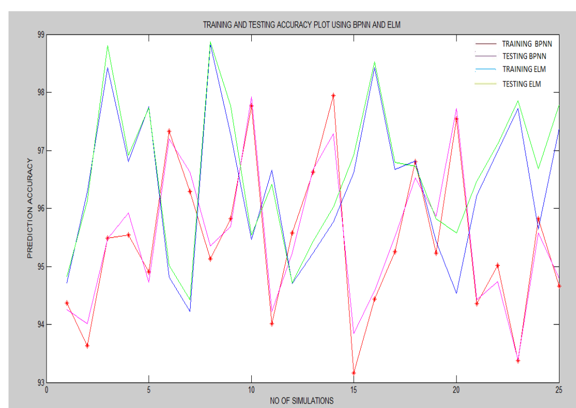

Fig7: Prediction Performance using BPNN & ELM

The mean absolute error MAE and the mean absolute percentile error MAPE has been calculated for both the method and tabulated below.

The **MAE** $= \frac{1}{n}\sum_{i=1}^{n} |Lactual_i - Lforecast_i|$ is tabulated below

Table.1: MAE for BPNN and ELM

MONTH	MAE(BPNN)	MAE(ELM)
MAR	108.428	51.571
DEC	69.427	41.572
SEP	152.851	70.574
JUN	206.814	161.347

The MAPE=$\frac{1}{n}\sum_{i=1}^{n}|\frac{Lactual_i - Lforecast_i}{Lactual_i}|$ x 100 is tabulated.

Table.2: APE for BPNN and ELM

MONTH	MAE(BPNN)	MAE(ELM)
MAR	3.5997	1.608
DEC	2.267	1.553
SEP	4.978	2.213
JUN	6.608	5.012

7. Conclusion

This paper has presented a new approach for short term load forecasting using extreme learning machine. The bottleneck observed in the use of BP algorithm can be overcome by use of ELM which shows faster convergence to global minima rather than local minima. The ELM approach shows better accuracy with very less training time and faster approach to obtain the optimized set of weights. Further studies will be focused at developing STLF models considering weather sensitive parameters for better convergence and accuracy.

References:

1. S.Mishra, S.K.Patra , Short Term Load Forecasting using neural network trained with genetic algorithm and particle swarm optimization,First International Conference on emerging trends in Engineering and Technology.
2. J. H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor, MI: Univ. Michigan Press, 1975.
3. H. Juidette and H. Youlal, “Fuzzy dynamic path planning using genetic algorithms”, Electron. Lett. vol. 36, no. 4, pp. 374-376, Feb. 2000.
4. G.-B. Huang, H.A. Babri, Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions, IEEE Trans. Neural Networks 9 (1) (1998) 224-229.
5. A. Savran, Multifeeback-layer Neural Network, IEEE Trans. Neural Networks Vol. 18 No. 2, (2007) 373-384.
6. Guang-Bin Huang,Extreme Learning Machines Learning Without Iterative Tuning A tutorial IJCNN2012/WCCI2012,Brisbane,Australia June 10 2012.