International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015) pp 34560-34564

A comparison of machine learning algorithms for electricity load forecasting

[1]Akshay P. Dubey, [2]Santosh Mohan Pattnaik, [3]Vijay Kumar Singore, [4]Sibani Palo, [5]Dr. Mathew M. Noel
[1,2,3,4,5] VIT UNIVERSITY, VELLORE
Email: [1]akshayprasaddubey@gmail.com, [2]pattnaik.santosh123@gmail.com, [3]vijay_singore@yahoo.com, [4]sibani.palo288@gmail.com, [5]mathew.m@vit.ac.in

Abstract- Electrical load forecasting is one of the important issues in maintenance and planning of the electrical load distribution. There are many techniques employed for electrical load forecasting in which Artificial Neural Networks are the very efficient and easy technique. Neural networks are trained with the previous datasets of inputs and outputs. Researchers have invented many algorithms to train the neural network for giving the best results. Since, electrical load forecasting is a very important task which should be done very accurately. Therefore, it is very necessary to decide the best training algorithm which will give the best result among all the training algorithms. In this paper, many training algorithms are used to train the network but only 3 overall best training algorithm results are compared with each other. These training algorithms are Conjugate Gradient, Bayesian Regulation and Resilient Backpropagation algorithm. Results are in terms of error, speed of training and accuracy of prediction of electrical which are compared for each training algorithm. In this, we have found that Resilient Backpropagation algorithm is the overall best training algorithm providing best results among all training algorithms.

Keywords- Electrical load forecasting, artificial neural network, training algorithms.

1. INTRODUCTION

Electrical loads may vary any time which affects to the maintenance and distribution of the electrical load system. Therefore, in planning, maintenance and distribution of electrical load, it is very important to know the electrical load accurately. Artificial intelligence is the developing research area with high efficiency. For forecasting, it is best suited technique. Many training techniques are invented by the researchers which are best qualified for the specific areas of application. There are many criteria which are fulfilled in several training algorithms.

[1] F.J. Marin, F. Garcia-Lagos, G. Joya and F. Sandoval presented a paper in which they trained with the recurrent network and showed that the best results are obtained. Many networks are also available to make the neural network. [2] K. Kalaitzakis, G.S. Stavrakakis, E.M. Anagnostakis presented a paper with the parallel processing approach and showed the best results for load forecasting. Load forecasting is being an important task which has to be completed very accurately. [3] Xinxing Pan, Brian Lee and Chunrong Zhang presented the load forecasting by several important algorithms which give the best results.

In this paper, the neural network is trained with many different training algorithms and finally, 3 best training algorithms are compared with their results as error, speed of training and the load prediction accuracy. Various weather conditions and type of the day are taken as the input dataset and the electrical current is taken as the output dataset. Network is trained with many training algorithms and results are compared with each other to find out the best training algorithm for electrical load forecasting. That algorithm should be used for the electrical load forecasting for achieving the best results of error, accuracy, stability of the result, fast training etc.

2. Artificial neural network

Artificial neural networks are the approximations to the human mind. It is having input layer of neuron, output layer of neurons and the hidden layers of neurons. Each neuron connects to the proceeding layer neurons in various patterns. Neurons are connected to each other by the synaptic weights, similar to the human mind. Neurons have their own threshold value to pas the signal. Input neurons take the inputs, multiply them with the weights, sums and passes to the next layer neuron. Thus, the networks learn from the past data and train itself for other data testing. Many training algorithms are invented to train the neural network which are best suited with specific areas of applications. An artificial neural network is shown in Figure. 1 with 3 input layer neurons, 2 hidden layer neurons and 1 output layer neurons.

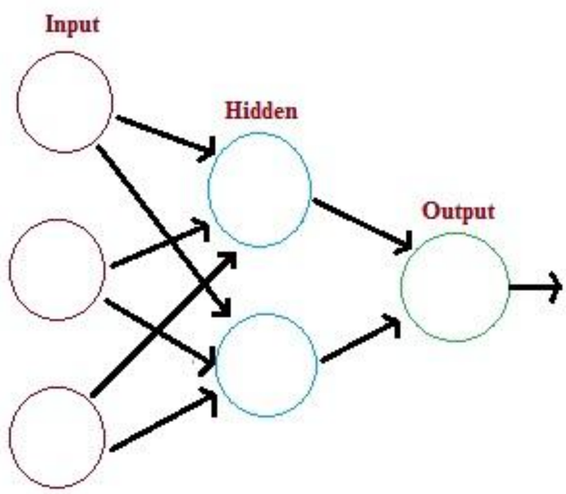

Figure 1.Artificial Neural Network

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015) pp 34560-34564

3. Training Algorithms

In this paper, many training algorithms are used to train the network but only 3 overall best training algorithm results are compared with each other. These training algorithms are Conjugate Gradient, Bayesian Regulation and Resilient Back propagation algorithms.

A. Bayesian Regulation Backpropagation

K.K. Aggarwal, Yogesh Singh, Pravin Chandra and Manimala Puri[4] presented the paper to estimate the code lines by using Function Points in which they proved that Bayesian Regulation is the best training algorithm.

This algorithm says that, in a finite dimensional system, every Reproducing Kernel Hilbert Space (RKHS) can be represented in terms of its specified featured map.

$$\mathrm{k}(\mathrm{x},\mathrm{x}') = \sum_{k=1}^{p} \phi^{i}(\mathrm{x})\phi^{\mathrm{i}}(\mathrm{x}')$$

$$f_{w'}(x) = \sum_{k=1}^{p} w^{i}\phi^{i}(\mathrm{x}) = (\mathrm{w},\phi(\mathrm{x}))$$

Also,

$$\|f_{w'}\|_{k} = \|w\|$$

Now, by building a Gaussian process, the final distribution function is as follows:

$$P(f,(X,Y)) \propto \exp\left(-\frac{1}{\sigma^2}\|f_{w'}(X) - Y\|_n^2 + \|w\|^2\right)$$

B. Conjugate Gradient Backpropagation

Martin F. Mollar[5] presented the paper and proved that Conjugate Gradient Algorithm is the fastest algorithm among all training algorithms. Conjugate Gradient Algorithm is the method of comparison of gradient descent with the step size which is optimal, to minimize the function of a linear system. The algorithm is as follows:

$r_0 := b - Ax_0$

$p_0 := r_0$

i:=0

repeat

$\alpha_i := \dfrac{r_i^T r_i}{p_i^T A p_i}$

$x_{i+1} := x_i + \alpha_i p_i$

$r_{i+1} := r_i - \alpha_i A p_i$

If r_{i+1} is very small, then stop the loop.

$\beta_i := \dfrac{r_{i+1}^T r_{i+1}}{r_i^T r_i}$

$p_{i+1} := r_{i+1} + \beta_i p_i$

i:=i+1

end repeat

So, the final result is x_{i+1}.

C. Resilient Backpropagation

Resilient Back propagation algorithm is used for supervised learning in feed forward networks. It takes each weight independently and checks for the sign change of partial derivative of net error.

If $\varepsilon^- < 1$, then update value is multiplied by ε^-.

If $\varepsilon^+ > 1$, then update value is multiplied by ε^+.

Finally, each weight is replaced by its own weight.

4. Training Data

In electrical load forecasting, it is very important to consider each and every factor that can influence the electrical load. The electrical load forecasting is done at the Men's hostel, VIT University, Vellore, Tamilnadu. In this paper, 9 weather data and 1 type of day is collected as the input data for 29 days per hour. Input data includes: Time, actual temperature, wind pressure, gusts, humidity, dew point, cloud cover, rain, day's length, type of day. Type of day is '0' if it is holiday and '1' if it is instructional day. Electrical load current is taken as the output data. The real time data is taken for 29 days day per hour.

Thus, the input dataset is of 10×696 matrix and the output dataset is of 1×696 matrix. Some of the samples of input and output dataset are shown in Table1.

Table 1: Training data samples

Time	0	1	2	3
Actual Temp(C)	27	27	26	26
Wind Pressure (KM/h)	11	9	9	7
Gusts (Km/h)	17	15	13	11
Humidity (%)	72	76	78	80
Dew Point(C)	22	23	22	22
Cloud Cover (%)	54	47	40	33
Rain (mm)	0	0	0	0
Day's Length (Hours)	12.29	12.29	12.29	12.29
Day Type [Instructional(1)/ Holiday(0)]	0	0	0	0
Hostel Load Current(A)	617	629	632	657

5. Implementation of Algorithms

In this paper, many algorithms are tested by training by it and the results are compared with each other. Now, first of all, a neural network is made defining number of input layer neurons, number of output layer neurons, number of hidden layer, number of hidden layer neurons, training function, threshold function etc. The neural network is implemented in MAT LAB through commands.

By best approximation, number of neurons in the hidden layers is taken as {(number of input neurons+ number of output neurons)*(2/3)}.

Following are the important commands used for the neural network;
net = new fit(P,T,S,TF,BTF,BLF,PF,IPF,OPF,DDF)

Table 2: Abbreviation of network development

P	Input vectors.
T	Target vectors
Si	Sizes of N - 1 hidden layers
TFi	Transfer function of ith layer.
BTF	Network training function
BLF	Weight/bias learning function
PF	Performance function
IPF	Input processing functions.
OPF	Output processing functions
DDF	Data division function
Eg: net=newfit(in,out,8,[ ],'trainbr')	

The training data is divided into 3 parts- Training data, testing data and Validation data. General ratio is 70:15:15 but we can program the ratio of each part of the training data as follows:
net.divide Param.train Ratio = 66/100;
net.divideParam.val Ratio = 17/100;
net.divide Param.test Ratio = 17/100;

Now, training has to be done. There are various functions available for the training. In this paper, neural network is trained with many training functions one by one and then compared with their results in terms of error, speed of training and the prediction results accuracy. Following is the command:
[net,tr] = train(net,inputs,targets);

Table 3: Abbreviations of training functions

gdm	Gradient descent with momentum Backpropagation
gda	Gradient descent with adaptive learning rule Backpropagation
gdx	Gradient descent with momentum and adaptive learning rule Backpropagation
rp	Resilient Backpropagation (Rprop)
cgb	Powell-Beale conjugate gradient Backpropagation
cgf	Fletcher-Powell conjugate gradient Backpropagation
cgp	Polak-Ribiére conjugate gradient Back propagation
scg	Scaled conjugate gradient Backpropagation
bfg	BFGS quasi-Newton Backpropagation
lm	Levenberg-Marquardt Back propagation
oss	One step secant Backpropagation
br	Bayesian regularization

Performance and regression plots are shown by the following commands:
Plot per f(tr)
plot regression(targets, outputs)
For prediction checking of the electrical load:
Test out = sim (net,test data)

6. Results

The neural network is always trained by the input and target data with many training algorithms and compared for the various results as the training speed, error and predicted data accuracy. Figure. 2 shows the Mean absolute percentage error (MPAE) as bar graph for various training algorithms trained for different numbers of neurons in hidden layer.

Figure. 3 shows performance graph in MPAE (%) for various training algorithms. Figure. 4 shows the time taken for network training and numbers of epochs for each training algorithm.

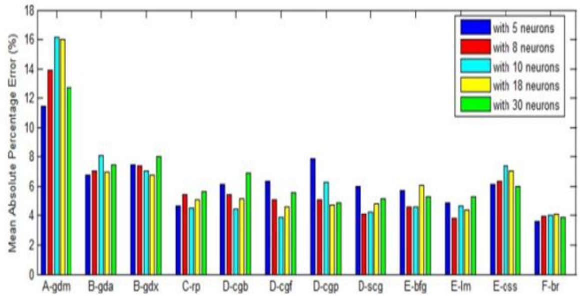

Fig 2. MPA Error for various training algorithm

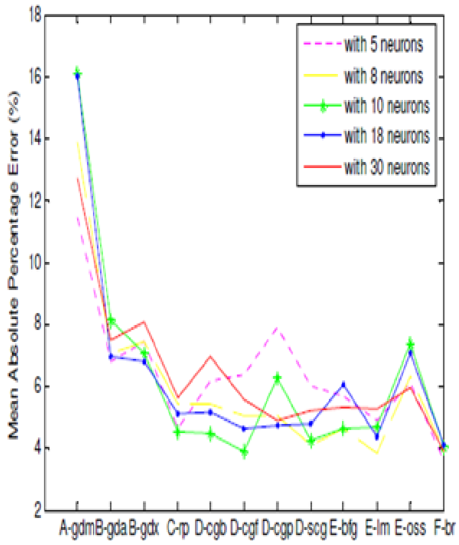

Fig 3. Performance graph for various training algorithms

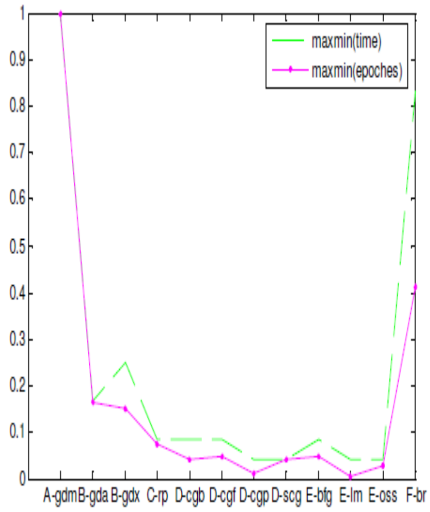

Fig 4. Number of epochs and time of training

Figure 5 shows training time for various training algorithms with variation in number of neurons.

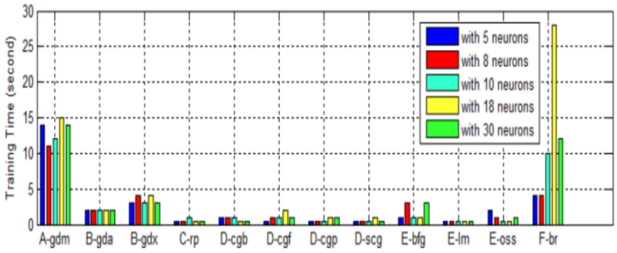

Fig 5. Training time for various training algorithms

7. Conclusion

The network is trained with 10 input parameters and 1 output parameter by 10×696 input data matrix and 1×696 output data matrix. The network is trained with many training algorithms and the result is displayed in terms of error, speed of the training (time) and the accuracy of the electrical load prediction. With this project, it is found that every training algorithm plays a significant role in specific areas of application. By comparing all the results, 3 algorithms are found giving the good characteristics among all training algorithms. These are, Conjugate Gradient, Bayesian Regulation and Resilient Backpropagation algorithm. Further, among these algorithms, Bayesian- Regulation Backpropagation Algorithm is little slow but overall it is the best training algorithm giving best results.

8. Acknowledgment

We thank to School of Electrical Engineering (SELECT), VIT University, Vellore for giving the opportunity to undertake this work. We also pay our gratitude to the Electrical Power house, VIT University, to give us support for this project. We are thankful our guide also for his valuable guidance to complete this project.

9. References

[1] F.J. Marin, F. Garcia-Lagos, G. Joya and F. Sandoval, “Global model for short-term load forecasting using artificial neural networks”, IEE Proc-Gener. Truiisni. Distrib, Vol. 149, Number 2, March 2002

[2] K. Kalaitzakis *, G.S. Stavrakakis, E.M. Anagnostakis, “Short-term load forecasting based on artificial neural networks parallel implementation”, Electric Power Systems Research 63 (2002) 185_ 196.

[3] Xinxing Pan, Brian Lee and Chunrong Zhang, “A Comparison of Neural Network Backpropagation Algorithms for Electricity Load Forecasting”, 978-1-4799-1135-1/13/$31.00 ©2013 IEEE.

[4] K.K. Aggarwal, Yogesh Singh, Pravin Chandra and Manimala Puri, “Bayesian Regularization in a Neural Network Model to Estimate Lines of Code Using Function Points”, Journal of Computer Sciences 1 (4): 505-509, 2005

[5] Martin F. Mollar, “A scaled conjugate gradient for fast supervised learning”, ISSN 0105-8517.

[6] Randal E. Bryant, Chase Hensel, Randy H. Katz and Erwin P. Gianchandani. “From Data to Knowledge to Action: Enabling the Smart Grid” Computing Community Consortium, Version 5, September 21, 2010.

[7] Milos Bozic, Milos Stojanovic and Zoran Stajic, “Short-term electric load forecasting using least square support vector machines” Facta Universitatis, Series: Automatic Control and Robotics Vol. 9, No 1, pp. 141-150, 2010

[8] B. J. Chen, M. W. Chang, and C. J. Lin, “Load forecasting using support vector machines: A study on EUNITE competition 2001,” Department of Computer Science and Information Engineering, National Taiwan University, Tech. Rep., 2002.

[9] I. Drezga, S. Rahman, Short-term load forecasting with local ANN predictors, IEEE Trans. Power Syst. 14 (1999) 3.

[10] B.-L. Zhang, Z.-Y. Dong, An adaptive neural-wavelet model for short term load forecasting, Electric Power Syst. Res. 59 (2001) 121_ 129.

[11] Akshay P. Dubey, K. Joshi, Y. Singh, Narendra S. Yadav, S. Pratush, P. Joshi, "Fuzzy logic based fault protection & fault classification system for induction motor", International journal of Electrical, Electronics and Data Communication, ISSN 2320-2084, Volume-1, Issue-7, September- 2013.

[12] Qi Wu, “Hybrid model based on wavelet support vector machine and modified genetic algorithm penalizing Gaussian noises for power load forecasts,” *International journal of expert systems with applications* pp.379-385, 2011.

[13] Akshay P. Dubey, Santosh Mohan Pattnaik, Dr. SaravanaKumar R., " Control & Operation of 4 DOF

Industrial Pick & Place Robot using HMI", Proceeding of Springer International Conference on Soft Computing, Spetember 20-21, 2015.

[14] Ying Chen, Peter B. Luh, Che Guan, Yige Zhao, Laurent D. Michel et.al., "Short-term load forecasting: Similar day-based wavelet neural networks," *IEEE Trans.on power syst.*, vol.25, pp.322-330, 2010.

[15] N. Amjady, and F. Keynia, "Short-term load forecasting of power systems by combination of wavelet transform and neuro-evolutionary algorithm," *International journal of energy*, vol.34, pp.46-57, 2009.

[16] Thulasiram, R.K.; Rahman, R.M.; Thulasiraman, P. "Neural network training algorithms on parallel architectures for finance applications", ICPPW' 03, 2003.