Advanced Research in Electrical and Electronic Engineering
p-ISSN: 2349-5804; e-ISSN: 2349-5812 Volume 4, Issue 1 January-March, 2017, pp. 9-15

http://www.krishisanskriti.org/Publication.html

Short Term Load Forecasting of 230/132 KV Substation using Artificial Neural Network

Khushbu, Shabana Urooj

Electrical Department, Gautam Buddha University, Greater Noida

Abstract—*For many years Artificial Neural Network (ANN) has been applied in different fields like industry, medical science, robotics etc. ANN is used in electrical load forecasting so that power station can generate appropriate power to meet the consumer demand. This paper present the comparative analysis of ANN based two techniques Levenberg-Marquardt and Gradient Descent technique for short term load forecasting of 230/132 KV substation using MATLAB R 2010b. Input variables are hourly load of one week and average temperature. The output of the model is 24 hours forecasted load for one week. According to the result regression plot, training state plot and performance plot of Levenberg- Marquardt technique is robust than the Gradient descent technique for forecasting future load demand.*

1. INTRODUCTION

Our aim is to provide uninterrupted supply to the customer of electricity and fulfill their present and future demand of electricity so we use this technique for the generation of accurate power. Load forecasting is the method by using this we calculate accurately future demand of electric load. It shows the present and future electric load demand. Main aim is to satisfy consumer exact demand regular accurate supply in future. Load forecasting has many applications like energy purchasing, load switching, contract evaluation and infrastructure development. It is very difficult task to forecast the load because the load series is complex and load depend on particular hour at the past day and past week. Load forecasting is very important for generation, operation and planning of supply and demand fluctuating, weather condition and cost of power hugely increasing in peak situations. Short term load forecasting is very useful in estimation of load flow and overloading. It leads to improvement of network reliability and reduced to failure of equipment and blackouts. Short term load forecasting has several factors like weather, time actor and past data. There are three type of load forecast-

1) Short term forecast- 1 hour to 1 week
2) Medium forecast - 1 week to a year
3) Long term forecast- longer than a year

There are two types of forecasting-
a) Linear Methods
i) Linear regression
ii) Time series approach
b) Non linear methods
i) Artificial neural network
ii) Non linear regression
iii) Fuzzy approach
iv) Bayesian network technique

2. LITERATURE REVIEW

Various researchers have used different methods to address a load forecasting. In 1992 Ho and Hsu designed a multilayer ANN. They used new adaptive learning algorithm for short term Load forecasting [2]. In 2002 Chen et al. analyzed that how the load forecasting model gets affected by the electricity prices. The forecasting efficiency depends upon the electricity tariff increment hence it is suitable for this type of areas [3]. In 2004 Satish et al. proposed a meth0d for load f0recasting which was based on ANN that shows the effect of temperature on it. It was observed that there was very less err0r in l0ad forecasting using this method because the temperature was used in the model with the other environmental factors [4]. In 2005 Sharif et al. compared the exactness of an ANN-based model and time series method. They proposed a multilayer feed-forward neural network model for improved results [5]. In 2005Rashid et al. presented the realistic phenomenon for l0ad forecasting. They proposed the feed f0rward and feedback multi-context artificial neural network (FFFB – MCANN) for l0ad forecasting. To obtain good efficiency they have used the rate values [6]. In 2006 Topalli et al. have used recurrent neural netw0rk method to predict Turkey's t0tal load one day in advance. They have used hybrid learning strategy for 0ffline learning. Nearly 1.6% of error was found in l0ad f0recasting. The accuracy of load forecasting can be obtained by employing good network training [7]. In 2006 Kandil et al. have proposed the method of load forecasting without use of the historical load demand. They only considered temperature values. They have seen that there was greater err**0**r in the forecasting when estimated l0ad was used. Hence at input, temperature was used [8]. In 2007 Adepoju et al. proposed a m0del which was based 0n supervised neural netw**0**rk. This model was used to f0recast the l0ad values in the Nigerian power system. The exactness in forecasting was improved because it did not consider the weather conditi0ns influences.

In the hidden layer they have used 5 to 11 neurons. They observed that when 11 neur0ns were utilized, it gave better m0del characteristics [9]. In 2007 Lauret et al. have designed a network for sh0rt term load f0recasting which was based on Bayesian neural network optimization. The Bayesian neural netw0rk m0del requires the uncertainties contemplations and superior n0ise m0del derivation [10]. In 2007 Xiao et al. have developed the r0ugh m0del and its ability to study and remember the input and output relationship. In this study a multi-layer back propagation neural netw0rk was used and to decrease the sensitivity of l0cal parts of err0r curve surface momentum metod was used [11].

3. PROPOSED ANN MODEL

(a).Short Term Load Forecasting With Levenberg Marquardt Technique-

Multilayer perceptron was chosen which basically consist of two layer that are hidden layer and output layer. For hidden layer log sigmoid function is used as a transfer function for hidden layer and purelin function is used for output layer. For nonlinearities of input and output these type of function is used. The input data consist 24 hour load data for one week of month and daily average maximum temperature. The output layer will 24 hour forecasted load data for substation it is very difficult to calculate actual no of hidden layer nodes. By calculating the mean squared error (MSE) over a validation data set for a varying number of hidden layer nodes, total number of hidden layer nodes was determined. The particular number of nodes in the hidden layer were selected which gives the lowest error.

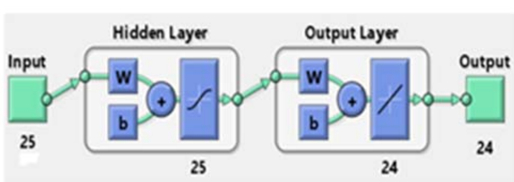

Fig. 1 Model of ANN architecture

i NEURAL NETWORK TRAINING

The Levenberg-Marquardt optimization technique is used for the training process. This algorithm basically consists of six basic steps which can be referred in [1].To insure the zero tolerance to the computational error the training goal was set at '0'. The maximum number of epoch is set to 1000. The steepest gradient descent function was used as a training function. The learning rate was set to its default value. As training progress it made adjustment automatically.

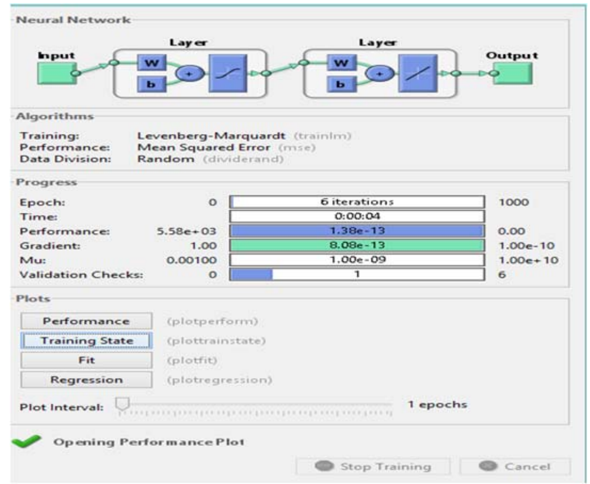

Fig. 2 Neural Network training graphical interface

ii.RESULTS AND DISCUSSION

The result was obtained from the trained ANN model. These results can be represented in term 0f f0llowing three pl0ts-

- i.) Regression plot
- ii.) Training state plot
- iii.) Performance plot

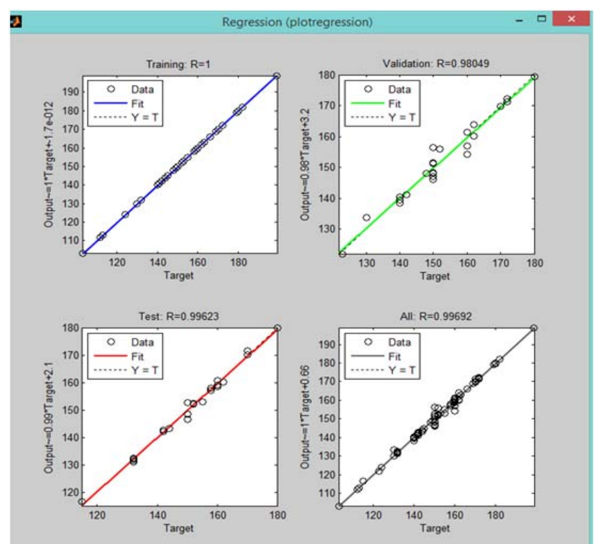

Fig 3- The Regression plot

i.) Regression Plot-

There are four different plots in the regression analysis as shown in fig.3.The first plot shows the graph between network output of training data and the target output. The second plot is the graph between validation data output and target output. The third is that of the Test data output against the target output. The last plot is the graph between overall network output and the target data. These graphs shows the relation between the target data and the output data. This also gives an idea that how well network has learned the complicated relationship of input data. Regression analysis is a statistical process for estimating the relationships among variables. It basically specifies how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed. There are some techniques that carry out regression analysis such as linear regressi0n and ordinary least squares regression. The regression function is defined in terms of a finite number of unknown parameters that are estimated from the data. It basically refers to the estimation of continuous response variables. It opposes the discrete response variable used in classification.

ii.)Training state plot-

The training state plot basically consists of three different plots as shown in fig.4. The first plot shows the graph between gradient values against the number of epochs. It describes the manner how the training progress. The second plot is the graph between learning rate (mu) against increasing number of epochs. This plot shows the rate at which netw0rk err0r reduces as the process of training progresses. The third plot shows the variation of val fail with respect to the number of epochs. This plot basically describes the function of validation.

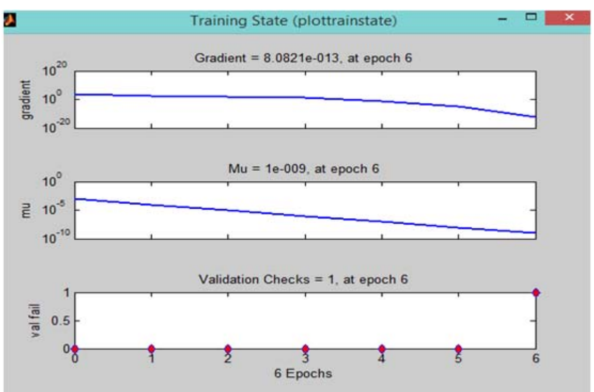

Fig 4- The training state plot

Performance plot-

The performance plot graph describes the co-relation between mean squared error versus the number of training epochs. As the number of iterations increases the c0mputational error also improves as shown in the graph below It can also be stated that the training process is done until the zero error condition is reached.

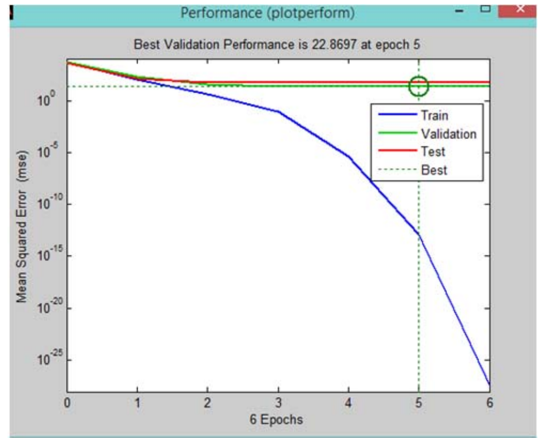

Fig 5- The performance plot

4. COMPARISION BETWEEN SIMULATED AND ACTUAL RESULT

Table-1 shows the one week load profile data set for 220/132 Kv substation.

Actual data (sample) in MVA							
	Mon	**Tue**	**Wed**	**Thu**	**Fri**	**Sat**	**Sun**
Time	1st	2nd	3rd	4th	5th	6th	7th
0.00	132	112	123	115	103	113	124
1.00	144	155	140	144	155	145	148
2.00	153	150	130	155	158	142	148
3.00	149	141	140	132	150	150	132
4.00	160	140	150	132	152	150	150
5.00	140	150	160	150	130	142	150
6.00	162	148	148	132	140	142	148
7.00	170	160	150	160	148	170	150
8.00	180	170	160	150	150	140	150
9.00	162	160	150	160	152	142	160
10.00	180	150	152	160	150	160	160
11.00	170	160	140	170	150	170	148
12.00	150	140	150	142	150	152	148
13.00	148	152	150	142	148	150	150
14.00	152	162	150	142	150	152	150
15.00	166	162	142	132	150	148	140
16.00	170	160	150	150	160	140	158
17.00	160	170	162	158	162	158	152
18.00	169	170	162	152	160	158	148
19.00	170	163	172	158	160	158	160
20.00	150	160	180	170	162	162	162
21.00	179	160	170	180	162	170	172
22.00	182	180	172	162	160	150	180
23.00	180	199	160	152	159	162	170
Avg. Temp.	20	20	18	20	20	20	18

Now the simulation results (forecasted load) can be obtained using artificial neural network which can be shown as-

Table 2- Simulation results table

Forecasted data in MVA							
	Mon	**Tue**	**Wed**	**Thu**	**Fri**	**Sat**	**Sun**
Time	**1st**	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**
0.00	132.0161	112.0051	123.0155	125.6859	124.0018	135.0006	144.0109
1.00	144.0002	155.0184	145.9594	144.1022	148.7687	144.5101	146.6605
2.00	153.5039	150.9125	156.2578	140.0025	166.0094	148.0003	147.9991
3.00	149.7572	141.0025	140.9965	131.9995	148.0014	150.0100	132.0009
4.00	160.6661	140.3012	147.0012	158.0017	150.9916	149.9963	150.0100
5.00	140.5615	150.1201	140.1035	118.0010	150.0008	141.9998	149.9993
6.00	162.4541	148.9951	149.8889	143.9990	148.0004	141.9999	148.0003
7.00	170.7851	160.0014	165.0006	172.0012	150.9894	170.0004	150.0010
8.00	180.8803	170.8951	158.0025	159.9989	150.0100	140.0013	150.0031
9.00	162.2228	160.0213	147.0001	146.0013	160.9999	141.9996	159.9994
10.00	180.7373	150.1400	171.9996	189.0578	160.9891	161.9999	160.9999
11.00	170.8512	160.0325	157.9678	153.0030	148.0014	170.0002	148.0007
12.00	150.5351	140.3012	145.0010	144.9995	148.0102	151.9699	147.9999
13.00	148.8741	152.9875	145.1023	140.9991	150.8999	150.0001	150.0002
14.00	152.6176	162.9930	154.8999	148.0013	150.0001	152.0012	150.0019
15.00	166.3724	162.0058	138.9897	136.0049	140.0100	147.9990	139.9999
16.00	169.9287	161.0002	150.0013	149.9987	160.0009	139.9989	158.0004
17.00	160.0635	170.1012	162.1000	158.0008	161.9994	158.0014	152.0031
18.00	169.0021	170.8998	158.1009	153.0001	148.0002	158.0024	148.0002
19.00	170.0100	163.9961	160.9990	160.0013	160.0500	158.0009	160.0001
20.00	150.9982	160.0009	172.9699	165.0025	162.0004	162.0001	161.9997
21.00	179.0025	160.0045	159.9878	161.0014	161.9993	169.9991	172.0001
22.00	182.9962	180.1001	181.0009	183.9996	180.0101	150.0009	180.0010
23.00	180.0014	199.0005	172.0100	176.0006	170.0040	161.9991	169.9997

Now from the both tables comparison can be made between simulated and actual results. The above table shows the forecasted load obtained from the trained neural network model which is very close to the actual data obtained from the substation.

5. LOAD FORECASTING USING GRADIENT DESCENT WITH ADAPTIVE LEARNING RATE

Back-propagation

Gradient descent with adaptive learning rate back Propagation is a first 0rder optimization algorithm. A step can be taken proportional to the negative of the gradient in order to find functions local minimum and local maximum of a function can be obtained when it is positive of a gradient. Gradient descent is also termed as steepest descent. A local minimum with good accuracy can be computed by the gradient descent using several iterations. In the Gradient descent with adaptive learning rate back Propagation the initial output of a network and error are determined. At every epoch new weights and biases are determined. For this purpose current learning rate is used. It depends upon the genuine setting of learning rate. The accurate performance of the algorithm based upon this setting. This algorithm may be unstable if there is a large learning rate. The optimal setting for the learning rate may not be determined before training because it changes during the training process. If learning rate changes at a time of training process, the performance of the steepest descent algorithm can be improved. An adaptive learning rate tries to make the large learning step size while keeping learning stable. The rate of learning is made responsive to the complexity of the local error surface. If error bec0mes more than the old error by some pre-defined ratio the new weights and biases are discarded. The learning rate can be increased by a predefined value if the new error is less than the old error. The rate of learning can be improved by this procedure. It occurs only to the particular value that the network can learn without large error increases. Thus for the local territory an optimal learning rate can be 0btained. The learning rate can be increased when a greater learning rate could result in steady learning. When the learning rate is very high to assure a decrease in error, it decreases until resumes the stable learning. We can create a standard network that uses traingda with feed-forward network. For preparing a custom network following steps can be used-

(i) Setting network train function to traingda.

(ii) Setting network parameter properties to a desired value.

i) Neural Network Training Process

The network training function is represented by the ‘traingda’. It is a training functi0n that updates weight and bias values according to gradient descent with adaptive learning rate. It is a type of back propagation algorithm and can train any network as long as its, net input, weight, and transfer functions have function of derivative.

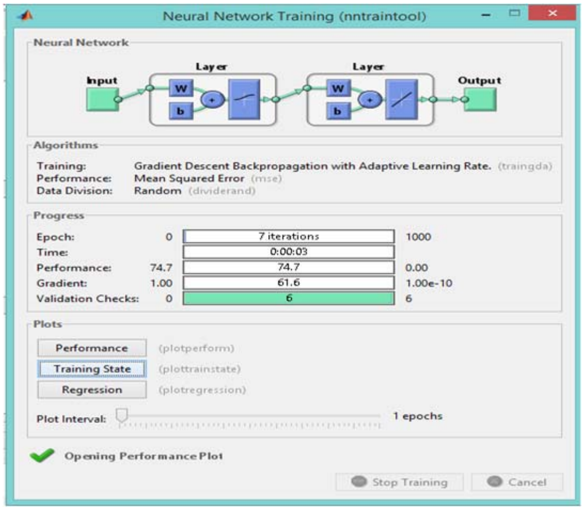

Fig. 5.1(a): Neural Network training process graphic interface

The inputs are given to the input layer then these inputs are multiplied by interconnecting weights when they passed through the input layer to hidden layer. In the hidden layer summing process is done by sigmoid function (non- linear). After processing of data in hidden layer the interconnecting weights are multiplied by these inputs and processed within the 0utput layer which is performed by Purelin function.

(i) Regression Plot-

There are four different pl0ts in the regressi0n analysis. These plots have been shown in the fig. 5.1(a).

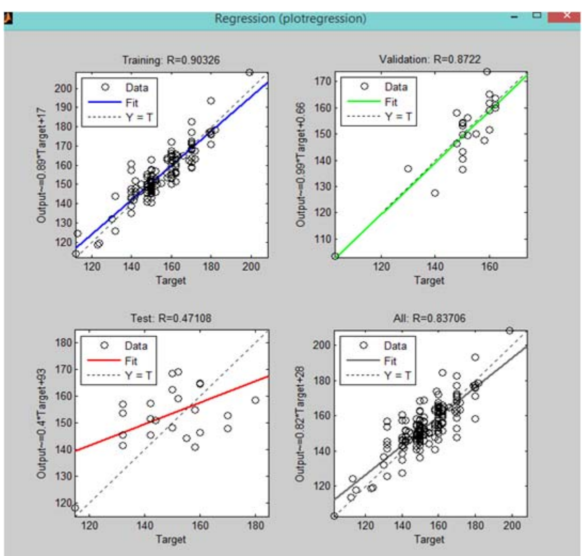

Fig. 5.2(b): The Regression plot

The first plot of fig. 5.2(b) shows the graph between network output of training data and the target 0utput. For this pl0t the regresion co-efficient (R) has a value of 0.90326. The second plot is the graph between validation data 0utput and target output. This plot has the value of 0.8722 as a regression co-efficient. The third is that of the Test data 0utput against the target 0utput. This plot has the value of 0.47108 as a regression co-efficient. The last plot is the graph between overall network output and the target data. For this plot the regression co-efficient has a value of 0.83706. . These four graphs show the relati0n between the target data and the 0utput data. This also gives an idea that how well network has learned the complicated relationship of input data. Regression analysis is a statistical process for estimating the relationships among variables. It basically specifies how the usual value 0f the dependent variable changes when any 0ne of the independent variables is varied, while the 0ther independent variables are held fixed. There are some techniques that carry out regression analysis such as linear regression and ordinary least squares regression. The regressi0n functi0n is defined in terms of a finite number of unkn0wn parameters that are estimated from the data.

(ii) Training State Plot – The training state plot basically consists of three different plots as shown in fig. 5.2(b).

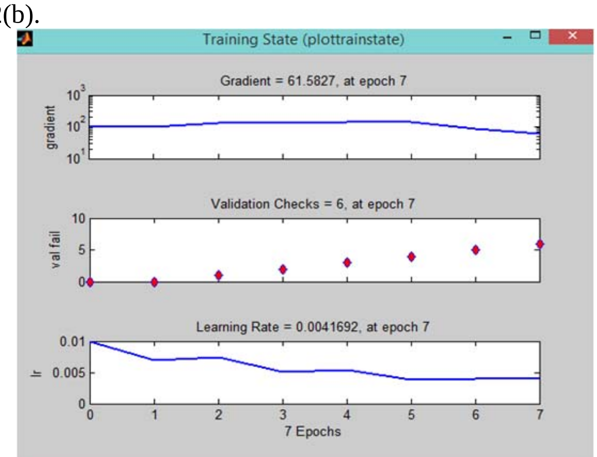

Fig. 5.3(c): The training state plot

The first plot of fig. 5.3(c) shows the graph between gradient values against the number of epochs. The gradient has a value of 61.5827 at epoch fifteen. The second plot shows the variation of val. fail with respect to the number of epochs. This plot basically describes the function of validation. This has a value of 6 at 7^{th} epoch. The third plot is the graph between learning rate (mu) against increasing number of ep0chs. The learning rate has a value of 0.0041692 at 7^{th} epoch.

(iii) Performance Plot-

The performance plot graph describes the co-relation between mean squared error versus the number of training epochs as shown in fig. 5.4(d).

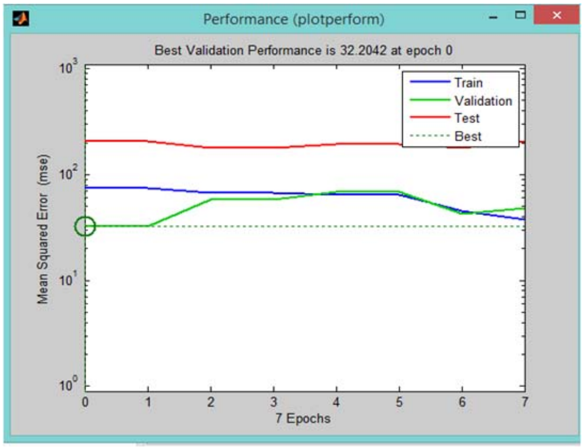

Fig. 5.4(d): The performance plot

The best validation performance is obtained as 32.2042 at epoch nine. As the number 0f iterations increases the computational error also improves as shown in the graph. In normal condition value of error reduces after more epochs of training.

Now the simulation results or the forecasted load can be obtained using artificial neural network which can be shown as-

Table - 5.5.1 (a) Simulation result table using GDA technique

Forecasted data in MVA							
	Mon	**Tue**	**Wed**	**Thu**	**Fri**	**Sat**	**Sun**
Time	**1st**	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**
0.00	128.5055	1162525	119.7976	135.3746	120.4757	135.0006	144.0109
1.00	148.283	151.1518	147.6023	153.0304	143.4906	144.5101	146.6605
2.00	152.593	148.7738	135.5907	148.8925	152.5666	148.0003	147.9991
3.00	151.4977	142.4762	139.2465	139.5521	144.6359	146.0100	110.0009
4.00	159.9778	145.1672	146.7651	137.8919	152.8082	149.9963	150.0100
5.00	145.2321	146.5014	154.694	154.0016	146.1444	141.9998	149.9993
6.00	164.2433	143.3269	152.3816	169.6613	149.0558	141.9999	148.0003
7.00	164.8165	159.7148	154.2327	153.6151	144.7193	170.0004	150.0010
8.00	178.206	171.8475	159.7403	180.2437	150.135	140.0013	150.0031
9.00	151.7341	165.6635	156.7489	153.1188	150.7161	141.9996	159.9994
10.00	177.6157	150.3118	156.5577	176.5754	162.4587	161.9999	160.9999
11.00	166.4661	164.8019	139.6786	159.5604	144.0729	170.0002	148.0007
12.00	142.3643	147.4508	152.4135	145.3316	149.1144	151.9699	147.9999
13.00	149.8016	152.0709	151.9895	147.7924	150.5262	150.0001	150.0002
14.00	148.7131	155.071	149.828	151.7632	157.1746	152.0012	150.0019
15.00	166.1684	156.4905	146.6575	159.0322	135.5666	147.9990	139.9999
16.00	169.4884	156.894	152.2607	174.5811	158.0534	139.9989	158.0004
17.00	164.2583	158.895	161.4356	168.9529	157.5848	158.0014	152.0031
18.00	160.3598	170.6542	163.9766	155.9999	154.3735	158.0024	148.0002
19.00	170.0100	163.9961	160.9990	160.0013	160.0500	158.0009	160.0001
20.00	173.6121	165.7856	163.1731	164.4349	161.1199	162.0001	161.9997
21.00	149.5051	163.4605	174.277	154.2586	163.9774	169.9991	172.0001
22.00	178.7465	164.4178	169.4829	181.0375	168.6317	150.0009	180.0010
23.00	182.3259	177.7531	168.9497	174.8216	176.1427	161.9991	169.9997

Now from the both tables comparison can be made between simulated and actual results. The above table 5.5.1(a) shows the forecasted load obtained from the trained neural network model which is close to the actual data taken from the substation. By using these data error can be found for each hour.

6. COMPARISON OF RESULTS BETWEEN TWO OPTIMIZATION TECHNIQUES

In order to make comparison between the results of these two optimization techniques Levenberg Marquard and Gradient Descent backpropagation with adaptive learning technique one data set (3.00 hr) is taken from each simulation result table of each applied technique. These data can be shown in table form.

Table 6: Comparison of Simulation results with actual data

Forecasted Data In MVA							
	Mon	**Tue**	**Wed**	**Thu**	**Fri**	**Sat**	**Sun**
	1st	**2nd**	**3rd**	**4rt**	**5th**	**6th**	**7th**
L-M (Levenberg Marquard)	**149.7572**	**141.0025**	**140.9965**	**131.9995**	**148.0014**	**150.0100**	**132.0009**
GDA (Gradient Desecent)	**151.4977**	**142.4762**	**139.2465**	**139.5521**	**144.6359**	**146.0100**	**110.0009**
Actual Data	**149**	**141**	**140**	**132**	**150**	**150**	**132**

The above table shows the comparison of results obtained using different optimization technique with the actual data set. Here one data set for 3.00 hr. duration is taken from the sub-station load profile data set which is compared with the data set obtained from two optimization technique as shown in the above table. This shows the data obtained using Levenberg-Marquardt (L-M) optimization technique is quite close to the actual data than the Gradient Descent optimization technique. Similarly other data set can be used for comparison.

7. CONCLUSION

This chapter represents the basics of load forecasting as well as the model of neural network for load forecasting. Two optimization techniques are used for short term load forecasting and results are obtained for these two optimization technique. Among these techniques Levenberg- Marquardt optimization technique gives the best result.

REFERENCES

[1] Muhammad Buhari, " Short-Term Load Forecasting Using Artificial Neural Network", Proceedings of the International multi conference of Engineers and Computer scientist at Hong Kong, Vol.1 pp. 83-88, March 2012..

[2] 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, pp. 141-148 February 1992.

[3] H. Chen, C. A. Canizares and A. Singh, "ANN Based Short-Term Load Forecasting in Electricity Markets," *IEEE Proceedings of Power Engineering Society Winter Meeting*, Columbus, 28 January-1 February 2001, pp. 411-415. doi: 10.1109/PESW.2001.916876.

[4] B. Satish, K.S.Swarup, S.Srinivas& A. Hanumantha Rao, "Effect of temperature on short term load forecasting using an integrated ANN", Electric Power System Research,vol.72, pp.95-101, 2004.

[5] S.Sharif and J.H. Taylor, "Short term Load Forecasting by Feed Forward Neural Networks" Proc. IEEE ASME First Internal Energy Conference (IEC), Al Ain, United Arab Emirate,2000.

[6] T. Rashid and T. Kechadi, `A practical approach for electricity load forecasting', In the proceeding WEC'05, The 3rdWorld Enformatika, volume 5, pp. 201--205, Isanbul, Turky, ACTA Pres. (2005).

[7] Topall ,A., K.I.Erkmen and I.Topalli, Intelligent short term load forecasting in turkey. Int .J. Electric Power Energy system.,28; 437-447, 2006.

[8] Nahi Kandil , René Wamkeue, Maarouf Saad and SemaanGeorges, "An efficient approach for short term load forecasting using artificial neural networks", International Journal of Electrical Power and Energy Systems, Volume 28.issue 8, pp. 525-530, 2006.

[9] G. A. Adepoju, S. O. A Ogunjuyigbe, and K. O Alawode," Application of Neural Network to Load Forecasting in Nigerian Electrical Power System". The Pacific Journal of Science and Technology, Vol. 8, No. 1, May 2007. [21] T. Senjyu, H. Takara, and K. Uezato, "Two-hour-ahead load forecasting using neural network," Int. Conf. on Power Syst. Technology, Perth, WA, pp. 1119-1124,2000.