ANALYSIS OF ELECTRICAL LOAD FORECASTING BY USING MATLAB TOOL BOX THROUGH ARTIFICIAL NEURAL NETWORK

Neeraj Pandey[1], Sanjay Kulshrestha[2], Manoj Kumar Saxena[3]
[1,2]Assistant Professor, Department of Electrical Engg., IPS-CTM Gwalior (M.P)
[3] Lecturer, Dept. of Electrical Engineering, Govt. Polytechnic College, Morena (M.P)
Email:neerajpandeyips@gmail.com[1], sk.kulshreshtha12@gmail.com[2], manojgwl@gmail.com[3]

Abstract

Load forecasting is a central integral process in the planning and operation of electric utilities. Load forecasting has become in recent years one of the major areas of research in electrical engineering. The main problem for the planning is the determination of load demand in the future. Because electrical energy cannot be stored appropriately, correct load forecasting is very essential for the correct investments. Effective forecasting, however, is difficult in view of the complicated effects on load by a variety of factors such as temperature, humidity. In this paper a three-layered feed forward neural network are trained by the Levenberg-Marquardt algorithm and a radial basis function using matlab programming and matlab tool-box. The Proposed neural network based model is used for forecasting next-week electricity prices. We evaluate the accuracy of the price forecasting attained with the proposed neural network approach, reporting the results from the electricity markets of India.

Keywords: price forecasting, neural networks, radial basis function, Levenberg-Marquardt Method, Matlab-Tool box

Introduction: Power load forecasting is one of the important works of power dispatching department. Improving the technical level of power load forecasting not only can accurately predict the demand of electricity market and convenient to power companies develop reasonable grid construction planning to improve the economic and social benefits of the system, but also be effective to predict the safety of power system operation and provide a reliable basis for the grid operation, maintenance and repair [1]. Various techniques of load forecasting had been reported for past decades. In summary, they can be classified as traditional methods based on mathematical models and soft-computing techniques. Hybrid of the referred methods [2-3] has also been achieving good performances. However, load forecasting is characterized with stochastic and uncertainty, so new techniques have to possess with high-level ability to represent and tackle various uncertainties. Thus Predicting future prices involved matching regional electricity demand to regional electricity supply.

The main problem for the planning is the determination of load demand in the future. Because electrical energy cannot be stored appropriately, correct load forecasting is very essential for the correct investments. Effective forecasting, however, is difficult in view of the complicated effects on load by a variety of factors such as temperature, humidity. Many short-term load forecasting methods have been developed including regressions methods, similar day approach, exponential smoothing; iterative reweighted least-squares, adaptive load forecasting; stochastic time series, ARMAX models based on genetic algorithms, fuzzy logic, neural networks and knowledge-based expert systems. Classical approaches are based on statistical methods. It assumes a stationary load series. The relationship between load demand and factors influencing the load demand however, is non linear therefore it is difficult to represent this complex non-linearity by conventional methods. Artificial intelligence methods have the ability to give better performance in dealing with the nonlinearity. A conventional ANN model sometimes can suffer from a sub optimization problem and over-fitting. These problems produce unsatisfactory forecasting accuracy. Selection of pre-training parameters and Network architecture significantly affects the performance and requires users to have in-depth knowledge of neural network methods.

In this paper a three-layered feed forward Neural Network is trained (1) first by the Levenberg-Marquardt algorithm (2) secondly by radial basis function. The proposed ANN structure has been prepared by using Neural Network Matlab Tool Box. We evaluate the results obtained by both the proposed approach and a comparative study has done. The inputs defined for the model are (1) maximum temperature of i^{th} day $T_{max}(i)$, (2) minimum temperature of i^{th} day $T_{min}(i)$, (3) average Temperature of i^{th} day $T_{ave}(i)$, (4) peak load of the i^{th} day $L_{max}(i)$ (5) peak load of the previous day $L_{max}(i-1)$ (6) peak load of seven days ago $L_{max}(i-7)$ and (7) peak load of the eight days ago $L_{max}(i-8)$.

To demonstrate the effectiveness of the proposed ANN approach, the both the ANN models have been trained and tested on actual load data and weather data for year 2014 to 2015 provided by i) Gwalior Region of Madhya Pradesh Madhya Kshetra Vidyut Vitaran Company Ltd. (MPMKVVCL) and Indian weather department, Indian Republic for peak load forecasting and ii) from Madhya Pradesh Power Transmission Company Ltd. (MPPTCL), Jabalpur for forecasting hourly load of M.P region.

The networks designed for one day ahead STFL have different numbers of input nodes, two hidden layers and 24 output nodes. To speed up the learning process adaptive learning rate is used. In the training process day type, the weather variables and load data of the past few days are used as input variables. Once trained, the designed ANN is capable of forecasting the next day peak load for twenty four hours accurately. To ensure the good training, the network was trained using large data set for 360 days and the unseen data of 65 days were used for testing the performance of trained network. Comparative study of different ANN structures has been done for different input parameters. It was observed that the designed ANN was capable of predicting the next day peak load quite accurately and RMS testing error was limited to 2.18 %.

It was observed in the present work that the results obtained by Levenberg-Marquard tmethod for the same ANN structure are better than results obtained by Radial basis functions. In addition for Radial basis function spread constant had to adjust to obtain better results. When all the parameters described above are taken as input the obtained results have minimum testing error.

An accurate forecast of electricity prices has become a very important tool for producers and consumers. In the short-term, a producer needs to forecast electricity prices to derive its bidding strategy in the pool and to optimally schedule its electric energy resources. In a regulated environment, traditional generation scheduling of energy resources was based on cost minimization, satisfying the electricity demand and all operating constraints. Therefore, the key issue was how to accurately forecast electricity demand. In a deregulated environment, since generation scheduling of energy resources, such as hydro and thermal resources, is now based on profit maximization, it is an accurate price forecasting that embodies crucial information for any decision making. Consumers need short-term price forecasts for the same reasons as producers.

It should be noted that price series exhibit greater complexity than demand series, given specific characteristics existing in price series. In most competitive electricity markets the series of prices presents the following features: high frequency, non constant mean and variance, daily and weekly seasonality, calendar effect on weekend and public holidays, high volatility and high percentage of unusual prices.

Price forecasting has become in recent years an important research area in electrical engineering, and several techniques have been tried out in this task. In general, hard and soft computing techniques could be used to predict electricity prices. The hard computing techniques, where an exact model of the system is built and the solution is found using algorithms that consider the physical phenomena that govern the process, include time series models , auto regressive — AR models and auto regressive integrated moving average — ARIMA models[4,5,6] . This approach can be very accurate, but it requires a lot of information, and the computational cost is very high. More recently, generalized autoregressive conditional heteroskedastic— GARCH models and the Wavelet-ARIMA technique have also been proposed.

The soft computing techniques, namely artificial intelligence techniques, do not model the system; instead, they find an appropriate mapping between the several inputs and the electricity price, usually learned from historical data, thus being computationally more efficient. In particular, neural networks approaches, which have been widely used for load forecasting with successful performance, are now used to predict electricity prices.

Neural networks are simple, but powerful and flexible tools for forecasting, provided that there are enough data for training, an adequate selection of the input–output samples, an appropriate number of hidden units and enough computational resources available[4,5]. Also, neural networks have the well-known advantages of being able to approximate any nonlinear function and being able to solve problems where the input–output relationship is neither well defined nor easily computable, because neural networks are data-driven. Three-layered feed forward neural networks are specially suited for forecasting, implementing nonlinearities using sigmoid functions for the hidden layer and linear functions for the output layer.
This paper proposes a neural network approach to forecast next-week prices in the electricity market of India. The Levenberg-Marquardt algorithm is used to train a three-layered feed forward neural network. Previously reported approaches to forecast prices in the electricity market of India were mainly based on time series models, namely the ARIMA technique. Neural networks approaches are comparatively easy to implement and show good performance being less time consuming.

2. Neural network approach

Neural networks are highly interconnected simple processing units designed in a way to model how the human brain performs a particular task. Each of those units, also called neurons, forms a weighted sum of its inputs, to which a constant term called bias is added. This sum is then passed through a transfer function: linear, sigmoid or hyperbolic tangent. Fig. 1 shows the internal structure of a neuron.

Multilayer perceptrons are the best known and most widely used kind of neural network. Networks with interconnections that do not form any loops are called feed forward. Recurrent or non-feed forward networks in which there are one or more loops of interconnections are used for some kinds of applications [8,9] .The units are organized in a way that defines the network architecture. In feed forward networks, units are often arranged in layers: an input layer, one or more hidden layers and an output layer. The units in each layer may share the same inputs, but are not connected to each other. Typically, the units in the input layer serve only for transferring the input pattern to the rest of the network, without any processing.

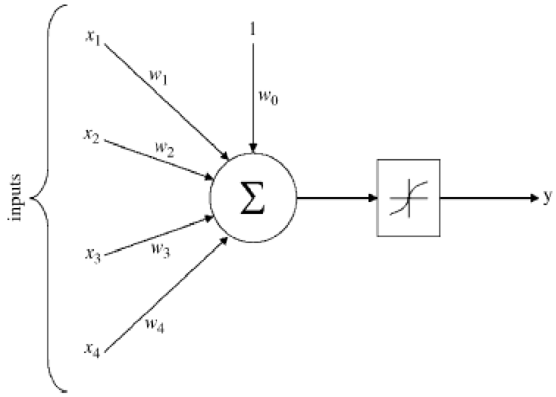


Fig. 1. Internal structure of a neuron.

The information is processed by the units in the hidden and output layers.

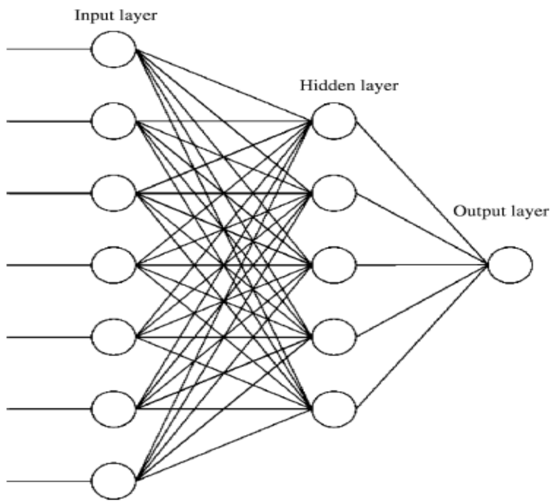


Fig. 2. Example of a three-layered feedforward neural network model with a single output unit.

Fig. 2 shows the architecture of a generic three-layered feed forward neural network model. The neural network considered is fully connected in the sense that every unit belonging to each layer is connected to every unit belonging to the adjacent layer. In order to find the optimal network architecture, several combinations were evaluated. These combinations included networks with different number of hidden layers, different number of units in each layer and different types of transfer functions. We converged to a configuration consisting of a one hidden layer that uses a hyperbolic tangent sigmoid transfer function, defined as:

$$f(s) = \frac{1-e^{-s}}{1-e^{-s}} \qquad (1)$$

Where s is the weighted input of the hidden layer, and $f(s)$ is the output of the hidden layer. The output layer has only one unit with a pure linear transfer function[10,11].

Forecasting with neural networks involves two steps: training and testing. Training of feed forward networks is normally performed in a supervised manner. One assumes that a training set is available, given by the historical data, containing both inputs and the corresponding desired outputs, which is presented to the network. The adequate selection of inputs for neural network training is highly influential to the success of training.

In the learning process a neural network constructs an input–output mapping, adjusting the weights and biases at each iteration based on the minimization of some error measure between the output produced and the desired output. Thus, learning entails an optimization process. The error minimization process is repeated until an acceptable criterion for convergence is reached [12].

The knowledge acquired by the neural network through the learning process is tested by applying new data that it has never seen before, called the testing set. The network should be able to generalize and have an accurate output for this unseen data[13]. It is undesirable to overtrain the neural network, meaning that the network would only work well on the training set, and would not generalize well to new data outside the training set [20]. Overtraining the neural network can seriously deteriorate the forecasting results.
Also, providing the neural network with too much or wrong information can confuse the network and it can settle on weights that are unable to handle variations of larger magnitude in the input data .

$$\Delta x = - [\Delta^2 v(x)]^{-1} \Delta v(x) \qquad (2)$$

where$[\Delta^2 v(x)$ is the Hessian matrix and $\Delta v(x)$ is a gradient vector. Assuming that V(x) is sum of square errors given by:

$$V(x)=\sum_{h=1}^{n} e_h^2(x) \qquad (3)$$

$$\Delta v(x)=2\,J^T(x)e(x) \qquad (4)$$

$$\Delta v^2(x)= 2J^T(x)J(x) + 2\,S(x) \qquad (5)$$

where e(x) is error vector and J(x) is jacobian matrix given by:

$$J(x)=\begin{bmatrix} \frac{\partial e_1(x)}{\partial x_1} & \frac{\partial e_1(x)}{\partial x_2} & \cdots & \frac{\partial e_1(x)}{\partial x_n} \\ \frac{\partial e_2(x)}{\partial x_1} & \frac{\partial e_2(x)}{\partial x_2} & \cdots & \frac{\partial e_2(x)}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_N(x)}{\partial x_1} & \frac{\partial e_N(x)}{\partial x_2} & \cdots & \frac{\partial e_N(x)}{\partial x_n} \end{bmatrix} \qquad (6)$$

and where $\boldsymbol{S}(\boldsymbol{x})$ is given by:

$$V(x)=\sum_{h=1}^{n} e_h(x)\Delta^2 e_h(x) \qquad (7)$$

Neglecting the second-order derivatives of the error vector, i.e. assuming that $\boldsymbol{S}(\boldsymbol{x}) \approx 0$, the Hessian matrix is given by:

$$\Delta v^2(x)= 2J^T(x)J(x) \qquad (8)$$

and substituting Eqs. (8) and (4) into Eq. (2) we obtain the Gauss–Newton update, given by:

$$\Delta x = -[\, J^T(x)\, J(x)\,]^{-1} J^T(x)e(x) \qquad (9)$$

The advantage of Gauss–Newton over the standard Newton's method is that it does not require calculation of second-order derivatives. Nevertheless, the matrix $\boldsymbol{J}T(\boldsymbol{x})\ \boldsymbol{J}(\boldsymbol{x})$ may not be invertible. This is overcome with the Levenberg-Marquardt algorithm, which consists in finding the update given by:

$\Delta \boldsymbol{x} = -[\boldsymbol{J}\,T(\boldsymbol{x})\,\boldsymbol{J}(\boldsymbol{x}) + \mu\,\boldsymbol{I}]^{-}1\boldsymbol{J}^{\boldsymbol{T}}(\boldsymbol{x})\,\boldsymbol{e}(\boldsymbol{x})$

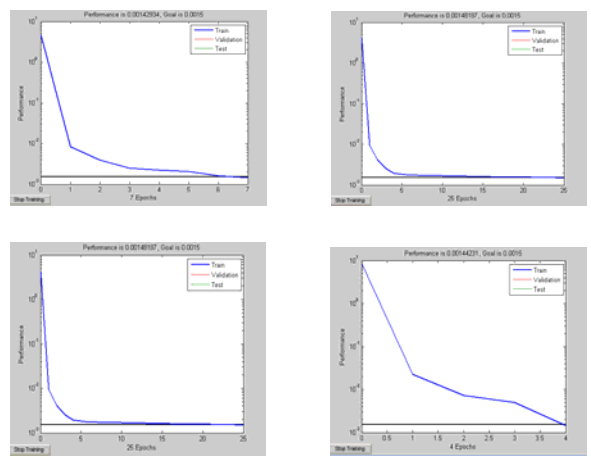
Where parameter μ is conveniently modified during the algorithm iterations.

Data selection and normalization:

Historical data from market, namely electricity prices and hours are input parameter to train the neural network in this paper. The transfer functions used for the hidden and output layers are, *tansig*, a hyperbolic tangent sigmoid transfer function with outputs between −1 and 1; *purelin*, a pure linear transfer function. Here data is normalized between 0 and 1 for training and testing purpose.

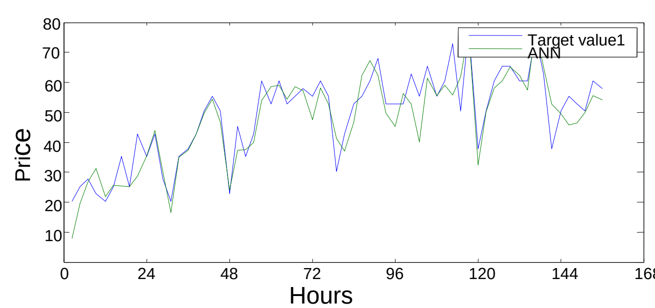
Case study: neural network approach based on Levenberg- Marquardt and radial basis function both, applied on historical data from electricity market. It was observed that the testing error in case of first method is less in comparison of second method.

Training results of different ANN Networks (6-10-24,6-15-24,6-30-24,6-40-24) with learning rate 0.05



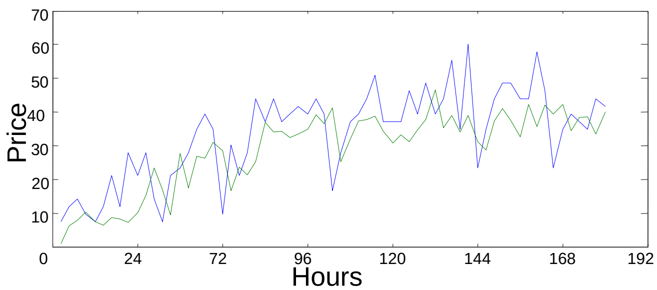
Result from Levenberg_Marquardt Method:

Training error(RMS value)	Testing error (RMS value)	Average percentage error
0.0349	0.0287	1.78%
0.0347	0.0294	2.01%
0.0348	0.0320	2.33%



Results from radial basis function neural network approach:

Training error(RMS value)	Testing error (RMS value)	Average percentage error
0.0601	0.0475	3.50%
0.0603	0.0424	3.09%
0.0600	0.0444	3.27%



Reference:

[1] Grzegorz Dudek, Pattern similarity-based methods for short-term load forecasting, Applied Soft Computing 37 (2015) 277–287, Elsevier, August 2015.

[2] Ming-Yue Zhai, "A new method for short-term load forecasting based on fractal interpretation and wavelet analysis", Electrical Power and Energy Systems 69 (2015) 241–245, Elsevier, February 2015.

[3] Luis H, Carlos B, Belen AJMC, Jaime S-EAJL, Joaquim M. A survey on electric power demand forecasting: future trends in smart grids, microgrids and smart buildings. IEEE Commun Surveys Tutorials 2014; 16(3):1460–95.

[4] Hernndez L, Baladrn C, Aguiar J, Calavia L, Carro B, Sanchez-Esguevillas A, et al. Artificial neural network for short-term load forecasting in distribution systems. Energies 2014;7(3):1576–98.

[5] S.Fan, R.J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model", IEEE Trans. Power Syst. 27 (1) (2012) 134–141.

[6] H. Kebriaei, B.N. Araabi, A. Rahimi-Kian,"Short-term load forecasting with a new non symmetric penalty function", IEEE Trans. Power Syst. 26 (4) (2011) 1817–1825.

[7] V. H. Hinojosa and A. Hoese, "Short-Term Load Forecasting Using Fuzzy Inductive Reasoning and Evolutionary Algorithms", IEEE Transaction on Powers Systems, Vol.25, February 2010.

[8] Ying Chen, Peter B. Luh, Yige Zhao, Laurent D. Michel, Matthew A. Coolbeth, "Short- Term "Load Forecasting Similar Day-Based Wavelet Neural Networks". IEEE transactions on power systems, Vol.25, No.1, February 2010.

[9] Mohammed K. Abd, "Electricity Load Forecasting based on Wavelet Neural Network Technique", American Journal of Applied Sciences 6 (5): 970-973, 2009, ISSN 1546-9239© 2009 Science Publications.

[10] F.J. Nogales, J. Contreras, A.J. Conejo, R. Esp´ınola, Forecasting next-day electricity prices by time series models, IEEE Trans. Power Syst. 17 (2) (2002) 342–348.

[11] L.A. Zadeh, The roles of soft computing and fuzzy logic in the conception, design and deployment of intelligent systems, in: Proceedings of the 6th IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 1997.

[12] J. Contreras, R. Esp´ınola, F.J. Nogales, A.J. Conejo, ARIMA models to predict next-day electricity prices, IEEE Trans. Power Syst. 18 (3) (2003) 1014–1020.

[13] C.P. Rodriguez, G.J. Anders, Energy price forecasting in the Ontario competitive power system market, IEEE Trans. Power Syst. 19 (1) (2004) pp. 366–374.

[14] R.C. Garcia, J. Contreras, M. van Akkeren, J.B.C. Garcia, A GARCH forecasting model to predict day-ahead electricity prices, IEEE Trans. Power Syst. 20 (2) (2005) pp 867–874.

[15] A.J. Conejo, M.A. Plazas, R. Esp´ınola, A.B. Molina, Day-ahead electricity price forecasting using the wavelet transform and ARIMA models, IEEE Trans. Power Syst. 20 (2) (2005) 1035–1042.

[16] G. Gross, F.D. Galiana, Short-term load forecasting, Proc. IEEE 75 (12)

[17] H.S. Hippert, C.E. Pedreira, R.C. Souza, Neural networks for short-term load forecasting: A review and evaluation, IEEE Trans. Power Syst. 16 (1) (2001) 44–55.

[18] A.J. Wang, B. Ramsay, A neural network based estimator for electricity spot-pricing with particular reference to weekend and public holidays, Neurocomputing 23 (1) (1998) 47–57.

[19] B.R. Szkuta, L.A. Sanabria, T.S. Dillon, Electricity price short-term forecasting using artificial neural networks, IEEE Trans. Power Syst. 14 (3) (1999) 851–857.

[20] F. Gao, X. Guan, X.-R. Cao, A. Papalexopoulos, Forecasting power market clearing price and quantity using a neural network method, in: Proceedings of the IEEE Power Engineering Society Summer Meeting, Seattle, WA, 2000, pp. 2183–2188.
[21] L. Zhang, P.B. Luh, K. Kasiviswanathan, Energy clearing price prediction and confidence interval estimation with cascaded neural networks, IEEE Trans. Power Syst. 18 (1) (2003) 99–105.
[22] L.A.F.M. Ferreira, T. Andersson, C.F. Imparato, T.E. Miller, C.K. Pang,
A. Svoboda, A.F. Vojdani, Short-term resource scheduling in multi-area hydrothermal power systems, Electr. Power Energy Syst. 11 (3) (1989) 200–212.
[23]J.P.S. Catal˜ao, S.J.P.S. Mariano, V.M.F.Mendes, L.A.F.M. Ferreira, Parameterisation effect on the behaviour of a head-dependent hydro chain using a nonlinear model, Electr. Power Syst. Res. 76 (6/7) (2006) 404–412.
[24] V.M.F. Mendes, S.J.P.S. Mariano, J.P.S. Catal˜ao, L.A.F.M. Ferreira, Emission constraints on short-term schedule of thermal units, in: Proceedings of the 39th International Universities Power Engineering Conference, Bristol,UK, 2004, pp. 1068–1072.

[25] O.B. Fosso, A. Gjelsvik, A. Haugstad, B. Mo, I.Wangensteen, Generation scheduling in a deregulated system. The Norwegian case, IEEE Trans. Power Syst. 14 (1) (1999) 75–80.