



Bilgi Yönetimi Dergisi

Cilt:3 Sayı:2 Yıl:2020

<https://dergipark.org.tr/pub/by>



Peer- Reviewed Articles

Research Article

Article Info

Date submitted: 02.12. 2020
Date accepted: 21.12. 2020
Date published: 31.12. 2020

Makale Bilgisi

Gönderildiği tarih: 02.12. 2020
Kabul tarihi: 21.12. 2020
Yayınlanma tarihi: 31.12. 2020

Keywords

*Electricity Load Forecast,
Artificial Neural Network,
Turkey.*

Anahtar sözcükler

*Elektrik Yük Tahmini, Yapay Sinir
Ağları, Türkiye.*

DOI numarası

10.33721/by.834285

ORCID

0000-0002-9228-7699 (1)

0000-0001-5513-9665 (2)

Electricity Load Forecasting via ANN Approach in Turkish Electricity Markets

*Türk Elektrik Piyasalarında YSA Yaklaşımıyla Elektrik Yükü
Tahmini*

Fazıl GÖKGÖZ

Professor Dr. at Ankara University, Faculty of Political Sciences Department of
Management, Quantitative Methods Division, fgokgoz@ankara.edu.tr

Fahrettin FİLİZ

Ankara University, Social Sciences Institute, Doctoral Program in Management,
fahrettinfiliz@gmail.com

Abstract

Forecasting electricity load has become the essential task for electric utilities, power plants and regulators. It is essential that electricity load forecasts, which are a vital necessity of energy policies, produce healthy and reliable results. Artificial neural networks (ANN) can learn complex and nonlinear relationships. This article introduces 400 different ANN models for electricity load forecasting. Model performances have compared with Mean Absolute Percentage Error (MAPE) and Diebold-Mariano (DM) test. The electricity load data used for this study range from 2014 to 2016. The variation in forecasting ability of ANN for different models has also discussed. Levenberg-Marquardt (LM) with log-sigmoid transfer function trains the best performance ANN model.

Öz

Elektrik yük tahmini yapabilmek, elektrik hizmetleri, enerji santralleri ve düzenleyiciler için gereklidir. Enerji politikalarını için büyük öneme sahip olan elektrik yükü tahminlerinin sağlıklı ve güvenilir sonuçlar üretmesi esastır. Yapay sinir ağları (YSA) karmaşık ve doğrusal olmayan ilişkileri öğrenebilir. Bu makale, elektrik yükü tahmini için 400 farklı YSA modelini tanımlamaktadır. Model performansları Ortalama Mutlak Yüzde Hata (MAPE) ve Diebold-Mariano (DM) testi ile karşılaştırılmıştır. Bu çalışma için kullanılan elektrik yükü verileri 2014 ile 2016 yılları arasında değişmektedir. Farklı modeller için YSA'nın tahmin kabiliyetleri tartışılmıştır. Log-sigmoid aktarım işlevine sahip Levenberg-Marquardt (LM), en iyi performanslı YSA modelini eğitir.

1. Introduction

Today our daily life mainly depends on the use of various forms of electricity. The rapid rise of industrialization within the last century has contributed to the growth of electricity consumption. With the liberalization of many electricity markets, utilities had to deal with the fact that accurate load forecasting became a necessity. Forecasting the electricity load is crucial for market participants, which are generation companies, energy transactions and planners. Market participants with good load forecasting may have efficient utilization of electric energy. Likewise, maintenance scheduling, economic load dispatches are generally carried out on the basis of the electricity load forecasting.

A proper load demand policy in the electricity industry is of the utmost important for the whole economy. Accurate load forecasting results in considerable saving for electrical utilities. 1% reduction in the average load forecasting error can save even millions of dollars (Hobbs et al. 1999, p.1342). In

*Bu makalenin araştırma ve yayın süreci "Araştırma ve Yayın Etiğine" uygun şekilde yürütülmüştür.

making the policy decision in competitive electricity markets need more algorithms-based tools. Turkish future electricity strategy is based on privatization. The underlying reason for this strategy is to create in confidence investor to be engaged in generation activities (Güney, 2005, p. 103).

Load forecasting task in the electricity market is difficult due to the complexity of the market. This complexity arises from the instant nature of the electricity, complex market design and frequent regulatory interventions. Artificial intelligence is a field of research that is relatively new but is growing in quality. Computational intelligence is widely used to refer to fuzzy system areas, swarm intelligence, evolutionary computing, and artificial neural networks. ANNs are one of those areas that are most frequently used in the forecast of electricity load. Due to its clear design, simple executor and good performance artificial neural networks have received more attention (Hahn et al., 2009, p.903).

In order to select the best training algorithms, transfer function and neuron number for the ANN, all possible combinations should be considered. This paper introduces 400 different ANN model for electricity load forecasting for the Turkish Electricity Market. Load forecasting models are compared according to their Diebold-Mariano (DM) test and MAPE. The paper is structured as follows: section 1 provides short review about load forecast; section 2 provides a short overview of artificial neural networks. Section 3 provides methods in the study and section 4 discusses implementations and empirical finding. Finally, results and some forward ideas are discussed in the conclusion.

2. Literature Review

2.1. General Electricity Load Forecasting Review

There are various types of load forecasting models based on time horizon, algorithms and architecture types. According to time horizon, we can characterize load forecasting methods as long-term, medium-term and short-term methods. Long-term load forecast methods are usually used to forecast next several years. Long-term load forecasting is attractive for electric utilities, which want to make investment decisions. Medium-term forecast characterizes for several months up to next year. These models are important for portfolio management. Short-term load forecast methods are usually used to forecast time horizons of hours, days or weeks. This type of models used by electric utilities for trading and availability of supply.

Broad ranges of models for load forecasting have been suggested. Electricity load forecasting is classified by traditional approaches¹ and computational approaches². In addition, hybrid models are applied for load forecasting by combining wavelet and neural networks (Kodogiannis et al. 2013, p. 30). Electricity load forecasting also is classified as; multiple regression, time series, adaptive load forecasting models, fuzzy logic, artificial neural networks and knowledge-based expertise (Alfares & Nazeeruddin 2002, p.23). Likewise load-forecasting techniques are classified as mathematical models, soft-computing techniques and hybrid of different techniques. Because of uncertainty on load characteristic, fuzzy system and neural network combination are proposed for load forecast (Lou & Dong, 2015, p. 35). Load forecasting MAPE results are range from 0.56% to 1.30% (Mandal et al. 2006, p.2128).

Studies on energy demand forecasting of Turkey have begun in the 1960s and starting from 1984 econometrics models have been applied for forecasting purposes (Kankal et al., 2011, p.1928). Fuzzy logic electricity model for Turkish market shows that electricity demand is strongly related to gross domestic product of the country. Additionally, short-term electricity forecasting with economic performance would provide reliable data for policy makers (Küçükali & Barış, 2010, p. 2440). Gross domestic product (GDP) per capita, population, inflation, unemployment count and temperature are used to forecast the Turkey annual gross electricity demand (Günay, 2016, p. 93).

Methods used in load forecasting models are artificial neural network, swarm optimization, genetic algorithm, harmony search algorithm and autoregressive integrated moving average (ARIMA).

¹ Traditional approaches include linear regression, nonlinear regression, time series models and Kalman filters.

² Computational approaches include neural networks, support vector machines and fuzzy systems.

Forecasted energy types are electricity, energy demand, energy dependency and energy consumption (Kankal et al. 2011, p.1929).

2.2. ANN Specific Electricity Load Forecasting Review

ANN models have received much attention because of their easy implementation, clear design, fault tolerance and good performance. Advantages of ANNs is the capability of learning, massively parallel computation and robustness in the presence of noise. However, ANN models have a disadvantage of over-fitting, under-fitting and local optimal solutions.

Load forecasting techniques are reviewed and evaluated according to different architectures, maximum 1,825 training samples maximum 365 test days (Hippert et al., 2001, p.47). In this study, 7,296 sample hour data are used for testing, 19,008 sample hour data are used for training. Testing and training data samples in these models are comparatively much higher than other studies.

In addition, ANN models show its superiority to statistical methods in load forecasting (Srinivasan et al., 1991, p.1173). Artificial neural network based models are used for predict electricity load for Egyptian Unified System and result in the maximum value of 2.2% MAPE (Osman et al., 2009, p.7). Multilayer feed-forward ANN model with Levenberg- Marquardt optimization training algorithm is used to train network. (Buhari & Adamu, 2012, p. 125).

3. Methods

The first step to be taken when designing an ANN forecasting model is to select an appropriate architecture. There are several neural network subtypes. Radial basis function networks, recurrent neural networks are used for forecasting. The feed-forward neural networks are generally used in load forecasting (Hippert et al., 2001, p.46). Levenberg- Marquardt (LM), Broyden, Fletcher, Goldfarb and Shannon (BFGS), Gradient Descent (GD) and Gradient Descent with Momentum (GDM) optimizer (training) algorithms were used to create different models. The neuron numbers used in the hidden layer changes step by step from 1 to 50 to determine minimum network error. The activity function for neurons must be differentiable and non-decreasing as mentioning the appendix section. In the current study, hyperbolic tangent sigmoid transfer function and log-sigmoid transfer functions were used in models.

The evaluation of the artificial neural network performance was based on MAPE results and DM test. The average marginal effects of hidden layer neurons also have been determined. Lastly, the effect of short-load forecasting on energy policy was discussed. Figure 1 shows the steps of this study in order.

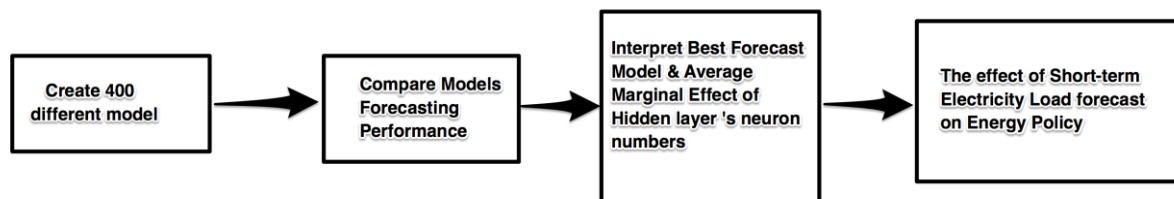


Figure 1. Steps of the Empirical Study.

4. Data and Models

4.1. Data

Electricity load data is received from Energy Exchange Operations Authority of Turkey (EPIAS). EPIAS's aim is to commence activities in electricity market under the supervision of Energy Market Regulatory Authority (EPDK). Data covers the 1,096 days between 01.01.2012 and 31.12.2014 and consist of 26,304 hours data, from this data 19,008 hours between 01.01.2012-03.03.2014 are used to train the model, 7,296 hours between 03.03.2014-31.12.2014 are used to test the model. Figure 2 shows training and testing data. The actual data and the forecasted data were compared through MAPE.

In addition to the MAPE test, we compared the accuracy of the forecasting results with The Diebold-Mariano (DM) test. DM test compares the forecast accuracy of two forecast methods. DM testing can discriminate the important distinctions between models (Diebold, 2007).

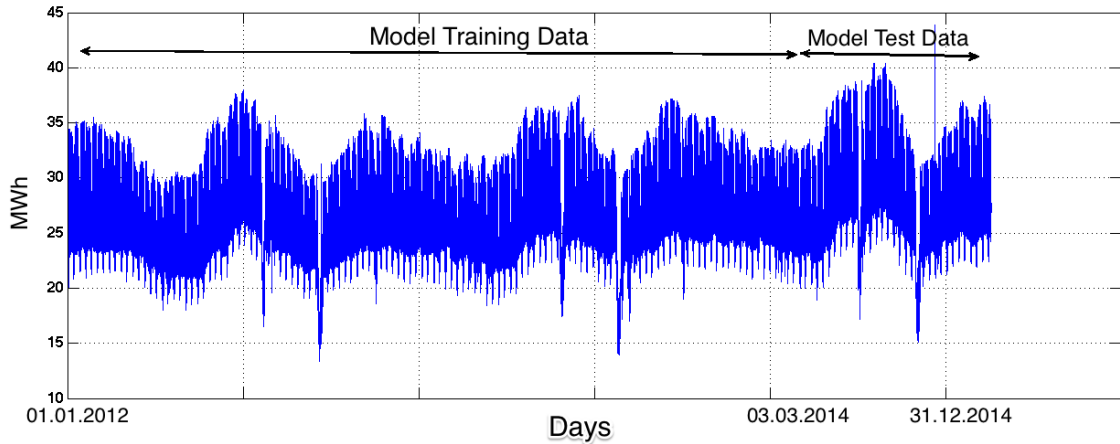


Figure 2. Model Training Data and Test Data.

4.2. Load Forecasting Factors

The selection of the input variables can significantly affect network performance. In the short term, load depends on the temperature, wind speed, the business activities such as weekdays, holidays and near-holidays. The optimal choice of input variables still remains an open question.

There is a common consensus that the temperature is the most significant impact in load forecasting (Hahn et al., 2009, p.903). The demand is high in cold climates because electric heating is common. Similarly, demand is high in hot weather, which can be attributed to air-conditioning compressors. These results in U-shaped function of the load with regard to the temperature (Hippert et al., 2001,p.50). Using hourly basis temperature increases the success of the model. Hourly temperature was used as input to the forecasting model for Turkey cities between days 01.01.2012 and 31.12.2014. The temperature information was acquired from www.wunderground.com. Population and industries in the city have affected electrical load consumption. For example, Istanbul, the largest city in Turkey has 16% load demand, Ankara, which is the capital of Turkey has 5.5%. Therefore, the temperature of the city and the electricity demand should be considered together. Figure 3 shows electricity load demand rates of Turkish cities.

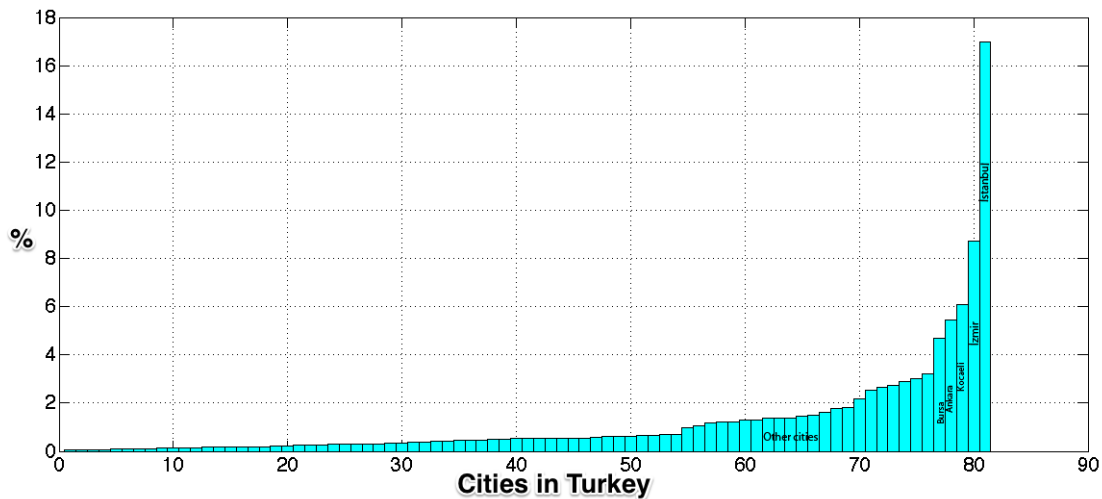


Figure 3. Turkey Electricity Load Demand Percentages by Cities.

Demand of load has a significant effect on electricity forecasting models. Models use the temperature with city-based electricity consumption. Electricity consumption has calculated by taking into consideration the load demand weighted calculated temperature. Equation 1 shows the load demand consumption weighted formula to calculate temperature.

$$\tau_m = \sum_{i=1}^{81} w_i t_i \tag{1}$$

i value shows cities in Turkey which has 81 cities, w_i is the electricity consumption percentage in the city, t_i the city temperature and m shows the day, τ_m reflects the load demand consumption weighted by temperature. Likewise, Table 1 shows input data statistics.

	Minimum	Maximum	Mean	Standard deviation
τ_m (°C)	-0,959	12,1922	5,4337	3,199
Load (MWh)	13,40	43,90	28,171	4,31

Table 1. Inputs Statistics.

4.3. Implementation of the Load Forecasting Model

Load Forecast Plan data was published by EPIAS (Energy Market Management Company) and was created by TEIAS (Turkish Electricity Transmission Company). Feed forward model, three-layered artificial neural network models are created. Gradient Descent (GD), Gradient Descent Momentum (GDM), BFGS and Levenberg - Marquardt (LM) algorithms are used in models in this study. Due to lack of precise rules to define neuron numbers in layers, varying neuron numbers between 1-50 are used in models. Neurons activation function is selected as log-sigmoid and hyperbolic tangent sigmoid transfer function. Four different algorithms (GD, GDM, LM, BFGS), hyperbolic tangent sigmoid transfer function and log-sigmoid transfer functions and 50 neurons formed 400 different load forecasting model. Table 2 shows the transfer functions and the number of neuron and training algorithms used in the electricity load forecasting application.

Algorithm	Transfer Function	Number of Neuron in Hidden Layer
GD	hyperbolic tangent sigmoid / log-sigmoid	1,2...50
GDM	hyperbolic tangent sigmoid / log-sigmoid	1,2...50
LM	hyperbolic tangent sigmoid / log-sigmoid	1,2...50
BFGS	hyperbolic tangent sigmoid / log-sigmoid	1,2...50

Table 2. Implementation of Load Forecasting Models.

Weekday, working or not working day (includes holidays and community festivals), previous week same hour electricity load, previous day same hour electricity load, 24-hour average electricity load and daily hour are model entries for load forecasting. Figure 4 shows model input, hidden layer and model output.

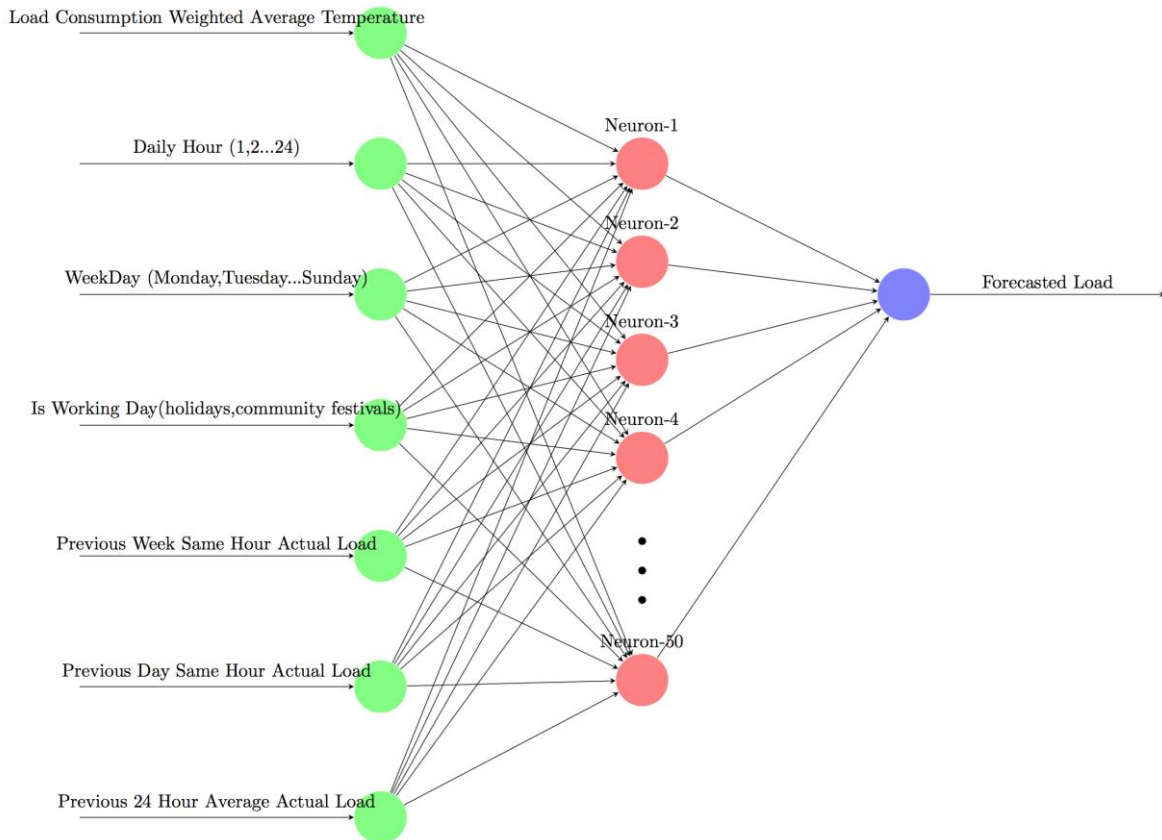


Figure 4. Electricity Load Forecast Neural Network Models.

4.4. Empirical Results

The performance of the 400 different models is evaluated according to models as model MAPE. In addition to the forecasting performance of the models, the training time of the models is also important. In this study, training time was not used to compare performance. But the increase in the number of neurons also increased the training time of the models. The most successful MAPE result was calculated as 1.86%. The Levenberg-Marquardt learning algorithm has been reached, this value with the log-sigmoid transfer function and 13 neurons in the hidden layer. Minimum, maximum and average MAPE values according to training algorithms are illustrated in Table 3.

	Min MAPE (%)	Max MAPE (%)	Mean MAPE (%)
GDM Log-sigmoid	3,1024	56,4818	22,1556
GDM Hyperbolic tangent sigmoid	3,1014	90,4378	40,0024
GD Log-sigmoid	3,9133	51,5335	40,0024
GD Hyperbolic tangent sigmoid	4,3468	58,0989	24,5190
LM Log-sigmoid	1,8600	3,9600	2,2040
LM Hyperbolic tangent sigmoid	1,9400	3,8400	2,2436
BFGS Log-sigmoid	1,9000	3,7600	2,1754
BFGS Hyperbolic tangent sigmoid	1,9400	4,3000	2,2264

Table 3. Load Forecasting Algorithms MAPE Results.

MAPE values based on the number of neuron change are given in Figure 5. Gradient Descent and Descent Momentum algorithms appear to have the higher MAPE values. As the number of hidden neurons increases, GD and GDM success decreases. GDM and GD training algorithm’s MAPE results are volatile compared to BFGS and Levenberg-Marquardt algorithms.

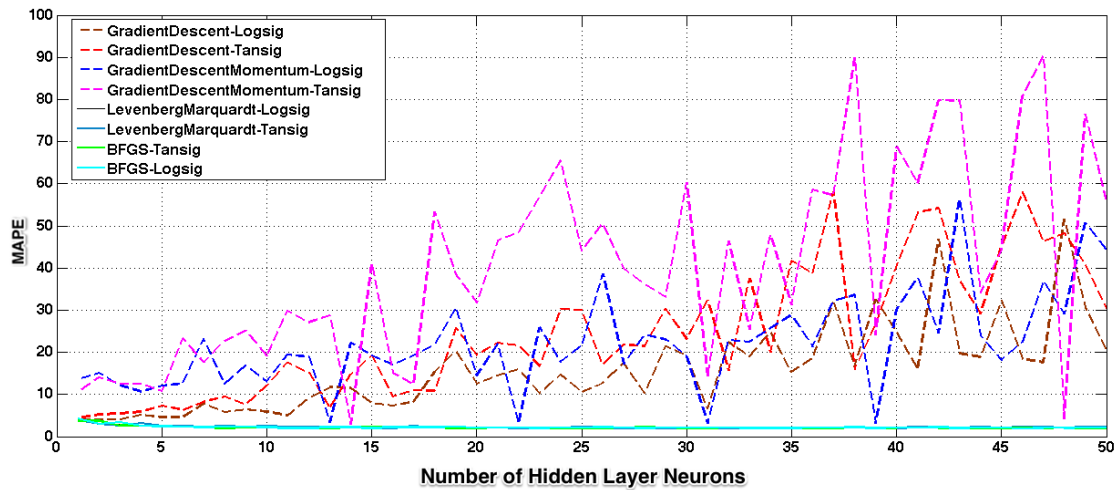


Figure 5. Load Forecasting Results for the All Algorithms.

	BFGS Hyperbolic tangent sigmoid	BFGS Log-sigmoid	LM Hyperbolic tangent sigmoid	LM Log-sigmoid	GD Hyperbolic tangent sigmoid	GD Log-sigmoid	GDM Hyperbolic tangent sigmoid	GDM Log-sigmoid
BFGS Hyperbolic tangent sigmoid	NaN	1,3729	-0,2297	1,0687	-6,3317	-4,9127	-6,8602	-6,8602
BFGS Log-sigmoid	-1,3729	NaN	-1,8141	-0,7176	-6,3335	-4,916	-6,8632	-6,9051
BFGS Log-sigmoid	0,2297	1,8141	NaN	1,361	-6,3325	-4,9121	-6,8612	-6,9047
LM Log-sigmoid	-1,0687	0,7176	-1,361	NaN	-6,3357	-4,9167	-6,8651	-6,9058
GD Hyperbolic tangent sigmoid	6,3317	6,3335	6,3325	6,3357	NaN	4,5911	1,7419	-4,9779
GD Log-sigmoid	4,9127	4,916	4,9121	4,9167	-4,5911	NaN	-5,9512	-5,9512
GDM Hyperbolic	6,8602	6,8632	6,8612	6,8651	-1,7419	5,9512	NaN	-

c tangent sigmoid								5,7523
GDM Log-sigmoid	6,9045	6,9051	6,9047	6,9058	4,9779	5,9512	5,7523	NaN

Table 4. DM Test Results.

The following empirical results are acquired from Table 4.

- According to DM test; between BFGS, LM compared values are less than 1,96, the zero hypothesis cannot be rejected at the 5% level of significance, so the BFGS and LM model’s forecasting performance is not significant.
- According to DM test; GD, GDM compared values are not less than 1,96, the zero hypothesis is rejected at the 5 % level of significance so the observed differences between GD to GDM, GD to BFGS, GD to LM, GDM to BFGS and GDM to LM are significant.

LM and BFGS algorithms have smaller MAPE values. Figure 6 shows LM and BFGS training algorithms MAPE changes according to the hidden layer neuron.

Neurons in hidden layers change from 1 to 15, which have a positive effect on the success of the model. There is no serious effect on the success of the model in more than 15 hidden layer neuron. Figure 7 shows the average marginal effect of the hidden layer neuron number.

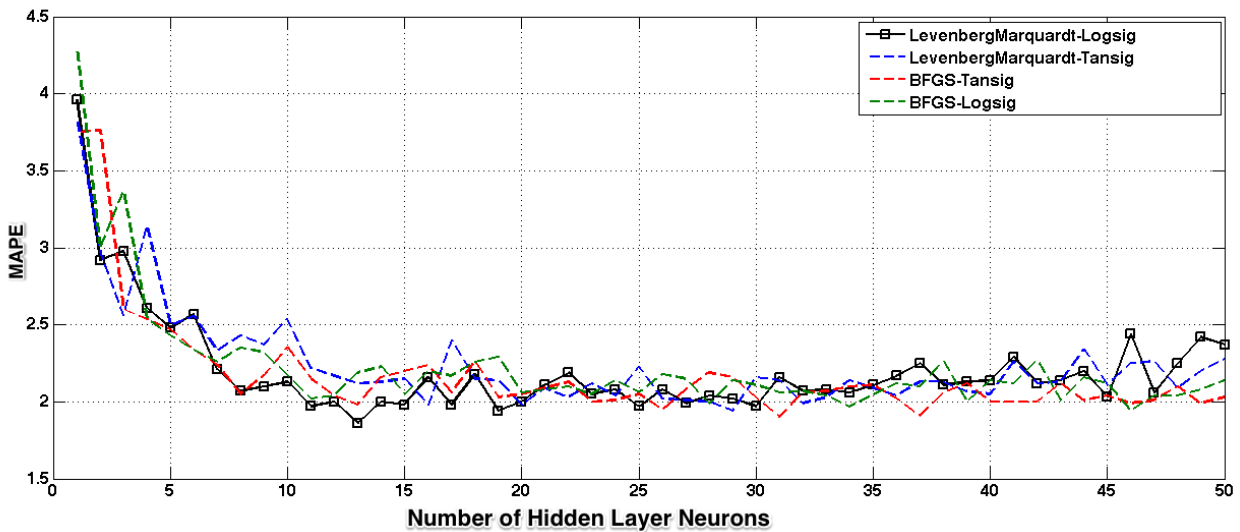


Figure 6. Load Forecasting Results for LM and BFGS.

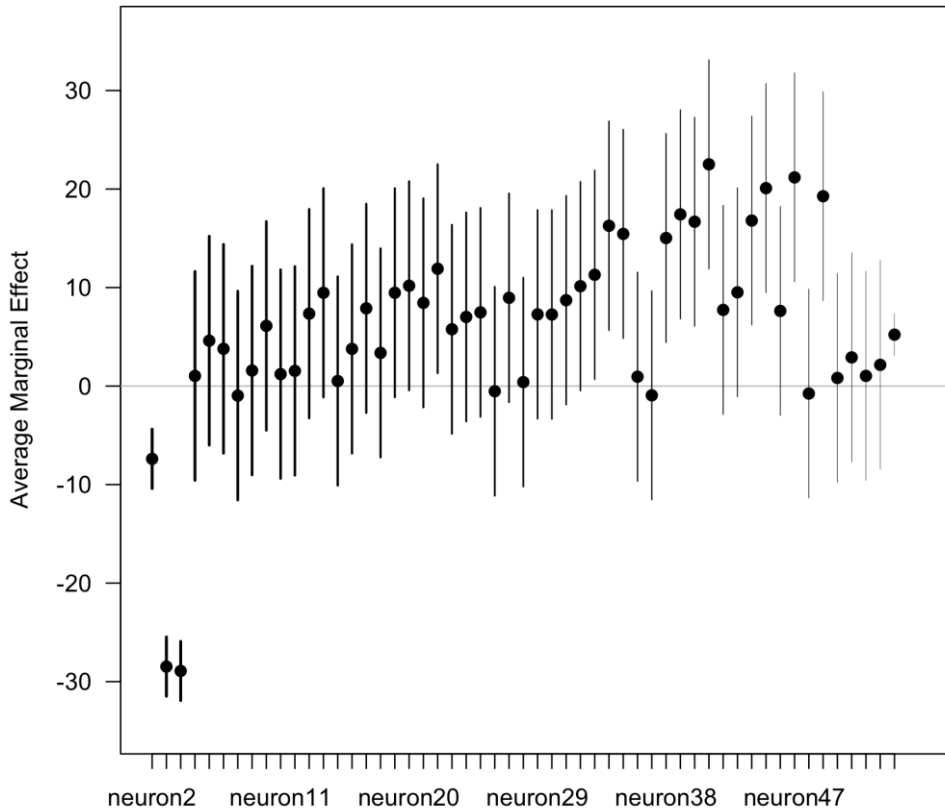


Figure 7. Average Marginal Effect of Hidden Layer Neuron Number.

LM training algorithm with 13 neurons in the hidden layer and log-sigmoid transfer function combination results in the most successful model. Actual electricity load data and forecasting electricity load are shown in Figure 8.

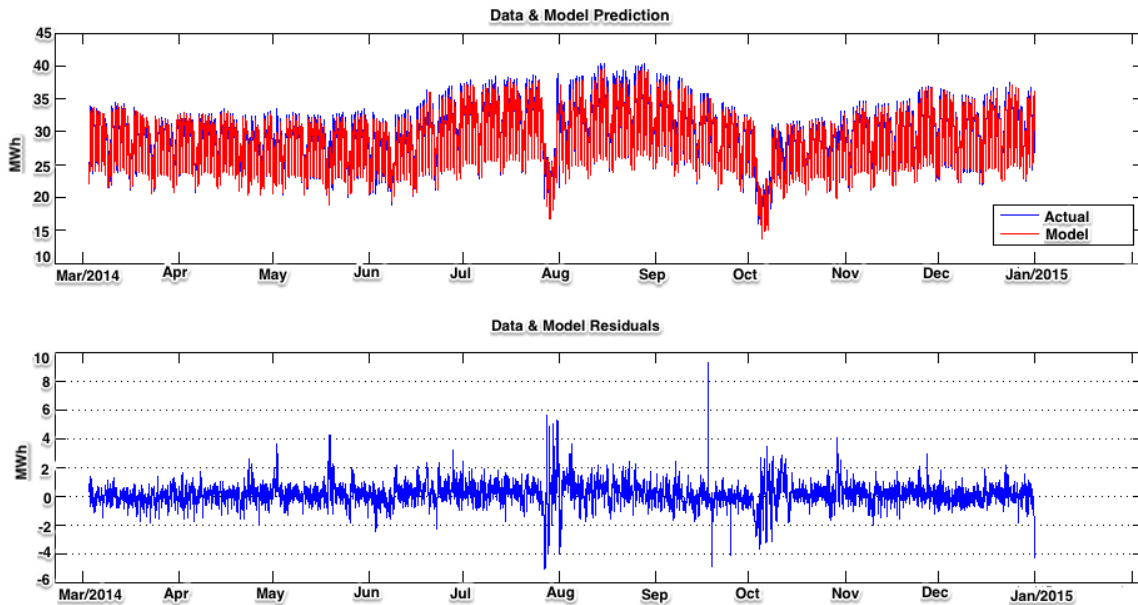


Figure 8. Actual Data and Model Prediction Data.

The average difference between the forecasted data and the actual data is calculated as 0,1373 MWh. The biggest difference between actual and estimated the amount of the electricity was observed on 18.09.2014 Thursday at 2 am. Load fluctuations greatly influence the forecast error. Figure 9 shows the change of Load Forecast Plan on 18.09.2014.

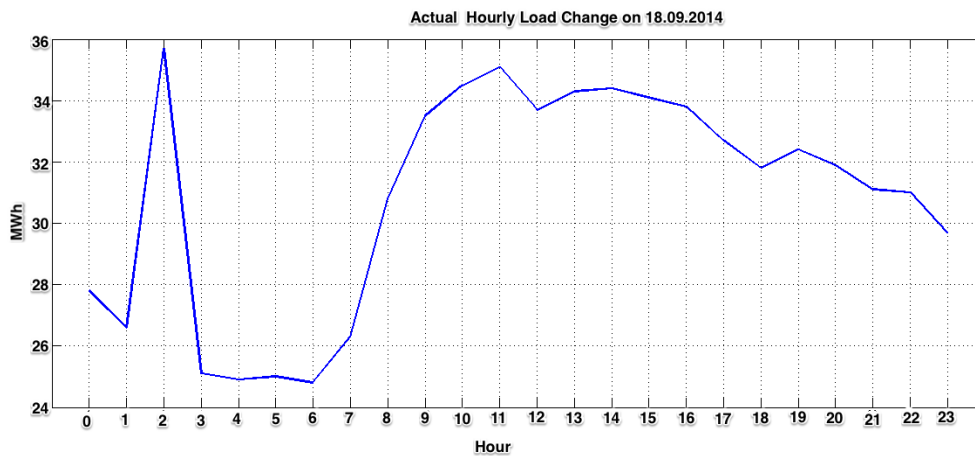


Figure 9. Load Changes as of 18.09.2014.

Forecasted electricity load value 26,4236 MWh and the actual load value 35,700 MWh at 2 AM on 18.09.2014. The reason for the maximum difference occurs at this hour made in the model sudden load value changes. Sudden increase or decrease of the load has a negative effect on the success of the models forecasting performance.

5. Conclusion

Effective implementation of electricity energy polices needs accurate load forecasting. Load forecasting contributes to planning and policy formulations. Modeling load forecasting plays a vital role for policy makers. Underestimation of the load would result in potential power outages, overestimation would result in idle capacity that means wasted resources. The decision makers which establishing a good load forecasting models needs to have efficient demand management. This paper presented comprehensive models for load forecasting with artificial neural networks approach in Turkish deregulated electricity market.

With changing algorithms, transfer functions and hidden layers neuron numbers, models have established. Model's performance has been compared according to MAPE values and DM test. The decision makers might use forecasts models to plan the investments and to achieve rational decisions.

LM and BFGS training algorithms give stable forecasting output. MAPE values in these algorithms change between 1.86% and 4.30%. Gradients Descent and Gradient Descent with Momentum methods MAPE values change between 3.10% and 90.43%. Levenberg-Marquardt and BFGS algorithms are well suited to neural network training according to MAPE and DM result.

In this regard, we may conclude that electricity load forecasting models established in this empirical study could have fruitful implications for decision makers.

Appendix A. Artificial Neural Network

Appendix A.1. Artificial Neural Training Algorithms

In this section, we provide an introduction into artificial neural networks. Inspired by biological systems, artificial neural network is a mathematical nonparametric and nonlinear model that maps neurons relationships while not exploring the internal processes. They capture functional relationships among the data, which has non-linear or difficult to describe as a relation. The characteristics of the artificial neural networks are adaptability, nonlinearity. These features make artificial neural network quite appropriate and useful for forecasting.

ANNs are applied in many fields such as medicine, defense industry, communications, finance, and econometrics. In an artificial neural network, the primary unit is artificial neuron. The single neuron is an information-processing unit, which takes input from other neurons and produces output. Single artificial neuron consists of five different components that are inputs, weights, activity function, bias and output. Figure 10 shows all elements of a single neuron.

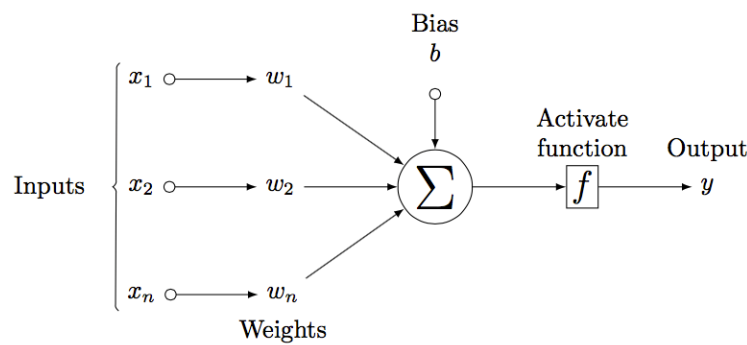


Figure 10. Mathematical Model of Neuron.

Input and output relationship in a single artificial neuron is as follows.

$$y = f\left(\sum_{i=1}^n x_i w_i + b\right) \tag{2}$$

x_1, x_2, \dots, x_n shows inputs, w_1, w_2, \dots, w_n are weights, b shows bias and f shows activity function and y shows output. Most preferably, the activity function is the sigmoid function. Usually, the cause of the preference of the sigmoid function in a neural network is its limited functionality. Limited increasing or a decreasing output characteristic of sigmoid function makes the stable behavior of the network. The sigmoid function is formulated as follows.

$$f(x) = \frac{1}{1 + e^{-\beta x}} \tag{3}$$

The ultimate goal of artificial neural networks is to minimize the network error, which is the discrepancy between the actual results and the result of the network. Network error between the results is formulated as follows.

$$\varepsilon = \frac{1}{2} \sum_{k=1}^n (o_k - t_k)^2 \tag{4}$$

n shows the total output number of the neural network. o_k shows the output of the network of each k , t_k indicates the target to be met for each k value. The network performance is determined by the difference between o_k and t_k values. The artificial neural network learning process is a nonlinear minimization problem in which weights of a neural network are iteratively changed to minimize the overall mean.

In general, there is no algorithm to provide the optimal solution for a nonlinear optimization problem. Backpropagation algorithm is a popular training method, which is mainly called as gradient steepest descent method.

Different algorithms have been suggested to enhance the training performance of neural networks such as gradient descent (Wang et al., 2016), conjugate gradients (Khadse, Chaudhari, & Borghate, 2016), Quasi-Newton (Likas & Stafylopatis, 2000) and Levenberg-Marquardt (Basterrech et al., 2011, p.130). The second order techniques are more effective techniques of nonlinear optimization. BFGS and LM are second-order methods used in this study.

The network weights are moved along the negative of the performance function gradient in standard backpropagation. Gradient descent algorithm starts at a random point in the neural network weight matrix for each input to change the network to produce the desired output on an algorithm. Gradient descent updated weight matrix calculated formulated as follows.

$$\Delta w_k = -\alpha_k g_k \quad (5)$$

Δw_k shows weight change in weight vector, g_k shows weight vector change at k , illustrates the learning coefficient α_k . Learning coefficient α_k should not be too large or small. Small α_k value slows the learning network (Ali et al., 2014, p.1022).

Gradient descent algorithm is a disadvantage of having a high learning time learning, and fluctuations at the local minimum (Azar, 2013). To prevent artificial neural networks from entering the network oscillation, momentum is added. p shows added momentum factor.

$$\Delta w_k = -\alpha_k g_k + p \Delta w_{k-1} \quad (6)$$

The Quasi-Newton method examines changes in weight vector and the second order derivatives (Fine, 1999, p.156). When compared to gradient descent method, quasi-newton approach more quickly to reach the desired weight vector. Gradient Descent and Gradient Descent Momentum uses first order derivatives. With Quasi-Newton method, weights of the network are calculated by the second-order Taylor series.

$$\Delta w H(w_i) = -\nabla f(w_i) \quad (7)$$

Δw is Newton weight step size, $H(w_i)$ is a Hessian matrix(second derivatives). Δw in the neural network is used to calculate the weights updates.

$$w_{i+1} = w_i + \Delta w \quad (8)$$

The obtained load value is compared with the error rate. In the case of network error is below a certain level successfully accepted and successful weight the network uses vector.

$$\|\nabla f(w_i)\| < \varepsilon \quad (9)$$

If the Hessian matrix (second-order derivative matrix) is assumed to be equal to zero if the determinant can be inverse matrix.

$$\Delta w = -\nabla f(w_i) H(w_i)^{-1} \quad (10)$$

The weight value of the network is calculated by a second order Taylor series and Newton weight change. There are different updated formulas of Quasi-Newton method such as Davidon Fletcher Powell (DFP), BFGS.

Levenberg-Marquardt algorithm is seen as a combination of Gradient Descent and Quasi-Newton method. Levenberg-Marquardt algorithm rather than calculate the Hessian matrix, he uses the approximate value of the Hessian matrix to accelerate the algorithm (Talaee, 2014, p. 699). In Levenberg-Marquardt algorithm Hessian matrix calculated by Jacobian matrix as shown below.

$$H = J^t J + \mu I \quad (11)$$

New calculated weight vector and the Hessian matrix is given by the following formula.

$$w_{k+1} = w_k - [J^t J + \mu I]^{-1} J^t e \tag{12}$$

w refers the weight vector matrix I refers unit matrix, combination coefficient , J refers Jacobean matrix, ε refers to the error vector. If big μ is selected the above formula acting like gradient descent algorithm if small μ is selected the above formula will act as Newton’s method (Kourentzes et al., 2014,p.14). Case default initial μ is 0.001 and default increase factor is 10 and default maximum is μ 1e10.

There are also algorithms not mentioned in this study such as the resilient methods, conjugate gradient. Table 5 shows algorithms to be used in this study.

Algorithm	Formula
GradientDescent(GD)	$\Delta w_k = -\alpha_k \cdot g_k$
GradientDescentMomentum(GDM)	$w_{k+1} = w_k - \alpha_k g_k + \rho \Delta w_{k-1}$
BFGS	$\Delta w = -\nabla f(w_i) H(w_i)^{-1}$
Levenberg-Marquardt(LM)	$w_{k+1} = w_k - [J^t J + \mu I]^{-1} J^t e$

Table 5. Training Algorithms.

There is a considerable amount of research and methods on neural network. Different of optimization algorithm are used to select network parameters to minimize artificial neural network error. Some of methods are the steepest descent, conjugate gradient, quasi-newton and Levenberg-Marquardt.

Neural network learning methods can be examined as heuristic optimization techniques and numerical methods. Heuristic optimization techniques are grouped in gradient descent, adaptive neural network, and gradient descent with momentum and resilient algorithms (Lahmiri, 2011, p15). Numerical methods are categorized in Quasi-Newton and Levenberg- Marquardt.

The quasi-newton method is often faster than conjugate gradient cases because it does not calculate the second derivative. It updates Hessian matrix the each iteration. Finally, Levenberg-Marquardt method includes the Quasi-Newton and the steepest descent method’s important features.

Appendix A.2. Artificial Neural Network Design

An artificial neural network design is absence of certain rules. Despite the many satisfactory characteristics of the neural network, building an artificial neural network for a forecasting problem has own difficulties. Neural networks have many satisfactory characteristics but neural network design for forecast problem is not an easy process. The design process of the artificial neural network has a major impact on the success of the network. The lack of precise rules on network design results in significant problems. Decisions regarding the network design can not be found due to the strict rules of artificial neural networks in the design process can be given by trial and error method (Haykin, 1999, p.72).

In general, the design process of the neural network can be grouped under the following figure.

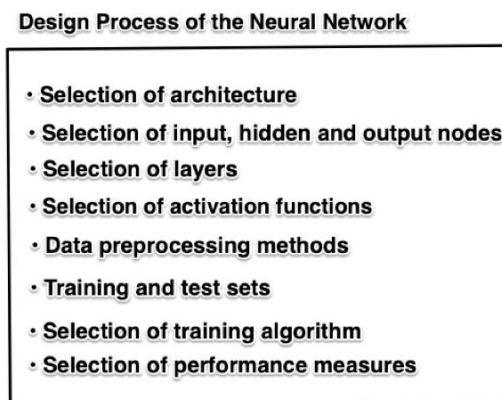


Figure 11. Design Process of the Neural Network.

Neural network design starts with finding problems inputs and output. In- puts and outputs of the network vary depending on the problem characteristics. Before starting the network design, inputs and outputs of the model must be analyzed. Input pre-processing can be done on the model inputs. Input data with carrying similar information can be simplified by pre-processing before entering the network. Data preprocessing will positively affect the performance of the network.

Selection of the network structure is a major step affects the success of the network. Feed forward network structure is preferred in general prediction models, classification models. In a recurrent network, the feedback problems are preferred. The different number of layers can be used in the network design process. An increased number of layers are proposed not to have a greater impact on the performance of the network (Zhang et al., 1998, p.44).

The neuron numbers in the layers should be determined intuitively. If the number of neurons in layers is too small, the input and output pattern can not be found by the network. As a result, the network may have difficulty in convergence during training. If the number of neurons is too large, the learning process will take long time and will impair the capacity of the neural network.

There is no learning method to provide the globally optimal solution for a nonlinear optimization. The back propagation algorithm, which is basically the steepest descent method of gradient, is currently the most popular training method. The BFGS and LM are efficient training methods. The transfer function should be differentiable and most papers used either the logistic or the hyperbolic tangent transfer functions.

Mean absolute percentage error (MAPE), the weighted mean absolute percentage error (WMAPE), the mean absolute error (MAE) and root mean square error (RMSE) is generally used to measure network performance. MAPE values are calculated using following equation where A_t shows the real value and F_t shows the estimated value.

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right| \quad (13)$$

Bibliography

- Alfares, H. K. & Nazeeruddin, M. (2002). Electric Load Forecasting: Literature Survey and Classification of Methods. *International Journal of Systems Science*. 33(1), 23-34.
- Ali, S. S., Moinuddin, M., Raza, K. & Adil, S. H. (2014). An Adaptive Learning Rate for Rbfn Using Time-Domain Feedback Analysis. *Scientific World Journal*. Doi: 10.1155/2014/850189.
- Azar, A. T. (2013). Fast Neural Network Learning Algorithms for Medical Applications. *Neural Computing and Applications Applications*, 23(3-4), 1019-1034.
- Basterrech, S., Rubino, G., Mohammed, S. & Soliman, M. (2011). Levenberg - Marquardt Training Algorithms for Random Neural Networks. *Computer Journal*. 54(1), 125-135.
- Buhari, M., & Adamu, S. S. (2012). Short-term Load Forecasting Using Artificial Neural Network . *Proceedings of the International Multi-Conference of Engineers and Computer Scientist*, 14-16.
- Diebold, F. (2007). Elements of Forecasting. *Thomson South-western*.
- Fine, T. L. (1999). *Feedforward Neural Network Methodology*. Springer.
- Guney, E. S. (2005). *Restructuring, Competition and Regulation in the Turkish Electricity Industry*, Tech. Tepav.
- Günay, M. (2016). Forecasting Annual Gross Electricity Demand by Artificial Neural Networks Using Predicted Values of Socio-Economic Indicators and Climatic Conditions: Case of Turkey. *Energy Policy*, 90, 92-101.
- Hahn, H., Meyer-Nieberg, S. & Pickl, S. (2009). Electric Load Forecasting Methods: Tools for Decision Making. *European Journal of Operational Research*, 199(3), 902-907.
- Haykin, S. S. (1999). *Neural Networks: A Comprehensive Foundation*. Pearson Prentice Hall.
- Hippert, H. S., Pedreira, C. E., & Souza, R. C. (2001). Neural Networks for Short-Term Load Forecasting: A Review and Evaluation. *IEEE Transactions on Power Systems*, 16(1), 44-55.

- Hobbs, B. F., Jitrapaikulsarn, S., Konda, S., Chankong, V., Loparo, K. A., & Maratukulam, D. J. (1999). Analysis of the Value for Unit Commitment of Improved Load Forecasts. *IEEE Transactions on Power Systems*, 14(4), 1342– 1348.
- Kankal, M., Akpınar, A., Komurcu, M. I. & Ozsahin, T. S. (2011). Modeling and Forecasting of Turkey's Energy Consumption Using Socio-Economic and Demographic Variables. *Applied Energy*, 88(5), 1927-1939.
- Khadse, C. B., Chaudhari, M. A. & Borghate, V. B. (2016). Conjugate Gradient Back- Propagation Based Artificial Neural Network for Real Time Power Quality Assessment. *International Journal of Electrical Power & Energy Systems*, 82, 197- 206.
- Kodogiannis, V. S. Amina, M., & I. Petrounias. (2013). A Clustering- Based Fuzzy Wavelet Neural Network Model for Short-Term Load Forecasting. *International Journal of Neural Systems*.
- Kourentzes, N., Barrow, D. K., & Crone, S. F. (2014). Neural Network Ensemble Operators for Time Series Forecasting . *Expert Systems With Applications*, 23(05), 1350024.
- Küçükali, S., & Barış, K. (2010). Turkey's Short-Term Gross Annual Electricity Demand Forecast by Fuzzy Logic Approach . *Energy Policy*, 38(5), 2438-2445.
- Lahmiri, S. (2011). A Comparative Study Of Backpropagation Algorithms In Financial Prediction. *International Journal of Computer Science, Engineering and Applications (IJCSEA)*, 1(4).
- Likas, A., & Stafylopatis, A. (2000). Training the Random Neural Network Using Quasi Newton Methods. *European Journal of Operational Research*, 126(2), 331-339.
- Lou, C. W. & Dong, M. C. (2015). A Novel Random Fuzzy Neural Networks for Tackling Uncertainties of Electric Load Forecasting. *International Journal of Electrical Power & Energy Systems*, 73, 34-44.
- Mandal, P., Senjyu, T., & Funabashi, T. (2006). Neural Networks Approach to Forecast Several Hour Ahead Electricity Prices and Loads in Deregulated Market. *Energy Conversion and Management*, 47(15-16), 2128-2142.
- Srinivasan, D., Liew, A., & J. S. Chen. (1991). A Novel Approach to Electrical Load Forecasting Based on a Neural Network, in: Neural Networks. *IEEE International Joint Conference on, IEEE*.
- Osman, Z. H., Awad, M. L., & Mahmoud, T. K. (2009). Neural Network Based Approach for Short-Term Load Forecasting. *Power Systems Conference and Exposition* .
- Talae, P. (2014). Multilayer Perceptron with Different Training Algorithms for Streamflow Forecasting. *Neural Computing and Applications*, 24(3-4), 695-703.
- Wang, L., Yang, Y., Min, M. R. & Chakradhar, S. (2016). Accelerating Deep Neural Network Training with Inconsistent Stochastic Gradient Descent. *Neural Networks*, 93, 219-229.
- Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with Artificial Neural Networks: The State of the Art. *International Journal of Forecasting*. 14(1), 35-62.