

# Short Term Load Forecasting System Using Artificial Neural Networks For Improved Demand Scheduling By Power Utilities.

Jubilee Mzingaye Sibanda

National University of Science and Technology Faculty of Applied Science

Jephias Gwamuri (✉ [jgwamuri@mtu.edu](mailto:jgwamuri@mtu.edu))

Great Zimbabwe University <https://orcid.org/0000-0003-1806-6799>

---

## Research Article

**Keywords:** load forecasting, power system planning, machine learning, Artificial Intelligence, SAPP

**Posted Date:** September 8th, 2021

**DOI:** <https://doi.org/10.21203/rs.3.rs-451487/v1>

**License:**  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

# Abstract

Load forecasting is a technique used by power utilities to predict electricity demand to maintain the balance between supply and demand. The problem comes when the power utilities draw more than the inadvertent power from the power pool. This necessitates the need for more accurate forecasting models. In this study, a short-term load forecasting system using artificial neural networks in MatLab was performed. The Levenberg-Marquardt optimization technique which has one of the best learning rates was used as a back-propagation algorithm for the Multilayer Feed Forward ANN model using MatLab® R2018a ANN Toolbox. Historical electricity load data obtained from a feeder line at the Zimbabwe Electricity Transmission and Distribution Company (ZETDC) Marvel 420 kV substation in Bulawayo Zimbabwe was used for the training, testing, and validation of the model. Results indicate that ANNs can forecast load with an accuracy of 6.71%. The results indicate that the proposed technique is robust in forecasting future load demands for the daily operational planning of power system distribution

## 1. Introduction

Electrical energy has to be generated whenever there is a demand for it. It is therefore imperative for the electrical power utilities that the load on their systems should be estimated in advance. For adequate electricity to be supplied to the customers, their load demand must be known. This estimation of load in advance is commonly called load forecasting. Due to a centralized power system and the continuous varying nature of the load, it is very difficult to balance demand and generation at all time [1]. Alternating current (AC) electricity cannot be stored and the science behind generation, transmission, and distribution of electricity requires that a stable or constant equilibrium be maintained between supply and demand in real-time. This can only be achieved by making reliable and accurate planning actions. If the system load forecast is overestimated, the system may overcommit the generation of power which will inadvertently lead to costly operation of the power system.

Zimbabwe's electricity generation is handled by Zimbabwe Power Company (ZPC) which is a subsidiary of Zimbabwe Electricity Supply Authority (ZESA). The utility has an installed capacity of about 2900 MW from two major sources for domestic electricity supply: which constitutes 58% thermal (coal-fired) and 37% hydro sources. In rural parts of the country, 80–90% of the people depend on wood fuel and kerosene for cooking and lighting [2]

Only 418 MW of the installed capacity was available on 21 January 2020 [3]. The difference is accounted for by age of the thermal plants and low water levels at Hydro-plants. Official figures from the Zimbabwe Power Company show that the country is currently generating an average of 418 MW from its five power stations against a daily peak demand of about 2100MW [3]. This means that ZESA now relies on regional imports to offset the deficit. Most of the units are now old and inefficient. For instance, the Hwange power station has an installed capacity of 920MW and averages 400–550 MW. The old coal-fired power stations in Harare, Bulawayo, and Munyati have a total installed capacity of 370 MW but most of the time they are not generating. The need for alternative energy supplies to meet the energy supply deficit cannot

be overemphasized. To meet the electricity supply deficit, ZESA has to import expensive power from its neighbours, mostly South Africa (Eskom) and Mozambique (Cahora Basa). The difference in demand is accounted for by under-budget generation and unplanned generator outages among other infrastructural reasons. The electricity supplies cannot meet demand as the electricity grid is in a poor state due to inadequate investment in the sector, leading to erratic supplies and load shedding. The danger with importing energy from neighbouring countries is drawing more than allowable inadvertent power of 95 MW from the Southern African Power Pool (SAPP) interconnected power system which attracts stiff penalties. ZESA has drawn more than the allowable advertent power of 95 MW from the SAPP system many times. As a result, this has reduced the revenue to import energy. Moreover, power outages continue to affect the economic performance of industries and services. Furthermore, poor forecasting methods have led to the distortion of the nominal frequency of 50Hz across the interconnected grid, and the power exchange threshold of 95MW has been violated several times while trying to restore the nominal frequency giving rise to the penalties. Proper techniques to forecast the load are hence imperative to eliminate these stiff penalties. This will enable power utilities to make economically viable decisions regarding future generation, transmission, and distribution. Also, the utilities will promote maximum utilization of power generation plants and avoid under-generation or over-generation and hence guarantee economic dispatch.

According to [4], there are 3 types of load forecasting which are short term load forecast which varies from 1 hour to one week and medium-term load forecast which varies from a week to usually a year, and lastly long-term load forecast which is longer than a year. Consideration of various factors is the prerequisite for accurate forecasting of the load. Whilst trying to keep the system reliability in reasonable tolerances, the forecasted system load must be met at the lowest possible cost which is a key result area in power system operation [5]. According to a review by [6], more than 113 different case studies across 41 academic papers have been used for the selection process. Factors such as time frame, inputs, outputs scale, data sample size, error type, and value were considered as criteria for the comparison. The review shows that despite the correlative nature of all reviewed models, the regression/multiple models are still very efficient and default to solving long and very long-term forecasting. [7] alludes that load forecasting at an individual household level is a challenging task that requires extracting load data directly and knowledge of the individual load profiles which are influenced by various factors. These factors include device operational characteristics, user behaviour, time of the day, and so on.

According to a review by [8], various methods can be used to solve SLTF which include Artificial intelligence (AI), Statistical techniques (non-linear regression, regression trees), and knowledge-based expert systems (fuzzy expert systems). Time-series methods treat the load pattern as a time-series signal with known seasonal weekly and daily periodicities. These periodicities give a rough prediction of the load at the given season, day of the week, and time of the day. The difference between the prediction and the actual load can be considered as a random signal [9]. A statistical approach to load forecasting using a regression model was done for the Zimbabwe Electricity Supply Authority (ZESA). The study aimed to construct an effective simplified econometric model that can be used to forecast the peak demand for

Loading [MathJax]/jax/output/CommonHTML/jax.js

The use of three different software environments, MATLAB,

SPSS, and Excel confirmed that they all have sufficient statistical capability to carry out reliable modelling. Results from MATLAB were most preferable because of their simplicity. The correlations investigation revealed that there was high collinearity with a Pearson correlation significant at the 0.01 level (2-tailed for: Peak demand with population and GDP per capita, maximum temperature, minimum temperature, and temperature range. The set of customized growth curves that align to different economic sectors and the constant percentage growth algorithms are used to align the electric forecast to the econometrics study [10]. However, according to [11] econometric methods have been long outmatched by other techniques due to lack of forecasting accuracy compared to forecast from simple mechanical schemes and autoregressions. Therefore, they are not advocated for in this application. Spatial forecasting has been also applied by Eskom, a South African power utility but due to this forecasting method, South Africa experienced serious power shortages in 2007. According to a review by [6], more than 113 different case studies across 41 academic papers have been used for the selection process. Factors such as time frame, inputs, outputs scale, data sample size, error type, and value were considered as criteria for the comparison. The review shows that despite the correlative nature of all reviewed models, the regression/multiple models are still very efficient and default to solving long and very long-term forecasting. Regional load forecasting involves predicting the amount of electricity that should be generated to supply specific kinds of consumers over a specific period and location.

The national grid which is based in the United Kingdom is collaborating with DeepMind®, a google owned Artificial Intelligence (AI) team. Deep mind is working on how to predict the power supply and demand peaks using statistics generated from the smart meters and weather stations within the UK. By using deep learning technology and machinery, demand and supply could be predicted and controlled in real-time as a result of load dispatch is optimized and operation costs are lowered [12]. According to [13] an ANN can be defined as a highly connected array of elementary processors called neurons. Artificial Neural Networks (ANNs) refer to a class of models inspired by the biological nervous system. The models are composed of many computing elements, usually denoted neurons; each neuron has several inputs and one output [14]. The figure below shows a single-layer perceptron.

In the above Fig. 1 [15], for one single observation,  $x_0, x_1, x_2, x_3 \dots x_n$  represents various inputs (independent variables) to the network. Each of these inputs is multiplied by a connection weight or synapse. The weights are represented as  $w_0, w_1, w_2, w_3 \dots w_n$ . Weight shows the strength of a particular node,  $b$  is a bias value. Mathematically, the sum of the products of weights and observations gives,

$$x_1w_1 + x_2w_2 + x_3w_3 + \dots + x_nw_n = \sum_{i=1}^n x_iw_i \tag{1}$$

Adding bias  $b$ , to

Eq. 1, gives:

$$v = \sum_{i=1}^n x_i w_i + b \quad (2)$$

According to [16], The most important unit in neural network structure is their net inputs by using a scalar-to-scalar function called "the activation function or threshold function or transfer function", output a result value called the "unit's activation". An activation function for limiting the amplitude of the output of a neuron. There are several activation functions which include Sigmoid Function, hyperbolic function, rectified linear unit activation function, and so on [17]. A Multi-Layer Perceptron (MLP) contains one or more hidden layers (apart from one input and one output layer), as shown in Fig. 2 below [18]. While a single-layer perceptron can only learn linear functions, a multi-layer perceptron can also learn non-linear functions, which makes it suitable for learning nonlinear, complex relationships hence their application in load forecasting.

In neural networks, to feed-forward is to give a certain input to the neural network, this can be historical load. The network will calculate the output by propagating the input signal through its layers. In other words, the output from one layer becomes the input to the next one, where the output from the last one is the final answer [19]. It has also been applied to load forecasting before.

## 2. Methodology

This section outlines the steps that were taken in designing the short-term load forecasting system that is based on artificial neural networks (ANNs). The assumption being that this system would lead to improved demand scheduling by power utilities and to alleviate the problem of drawing more than the inadvertent power from the interconnected system. Quantitative forecasting was applied because the two conditions that favor quantitative methods were met. These conditions are that the numerical data about the past was available and it was sensible to expect that some parts of the past patterns assume were to continue into the future.

Past ZETDC electricity load data from Marvel 420kV Substation was used as part of the inputs to the neural network. For ethical reasons, permission to access the ZETDC data was sought from the general manager of the company's Western region branch. The physical quantity that was in the past data sheets is the average electrical power and reactive power. It was measured using statistical energy meters which are programmable. The data from these meters was downloaded by special meter readers from the commercial engineering department of the power utility. The xlsx format was chosen over the csv format to preserve some features of the document that could get lost if saved in the csv file format. Data cleaning is the process of preparing data for analysis by removing or modifying data that is incorrect, incomplete, irrelevant, duplicated, or improperly formatted [20]. Pre-processing was done to enhance the forecasting model performance. Outliers, missing values, or any irregularities were removed since during Loading [MathJax]/jax/output/CommonHTML/jax.js ch defective data, and this could degrade the performance of

the model obtained. Outliers are data points that are dissimilar to the remaining points in the data set. If there is no correlation between energy consumption and the factors driving the consumption, the data point is considered an outlier if no other cause is identifiable [21]

## 2.1 Designing of the artificial neural network

MatLab is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment, where problems and solutions are expressed in familiar mathematical notation. Typical uses include application development including graphical user interface building. It was hence used as the prediction software. Predictive analytics is the use of statistical techniques like predictive modelling, machine learning, and data mining based on historical data to predict what will happen in the future.

The formulation of the MatLab code was done sequentially. The flow chart in Fig. 3 shows the general steps that were taken to create and evaluate the network. The network chosen for this application is the nonlinear autoregression with the exogenous input.

### 2.1.1 Creation and Definition of Data variables

- Data variables and training function were defined. For this application, an external time series data together with past load (electricity consumption) data values to be predicted were used as input fields. The data variables were further split into arrays containing 1x8760 columns of the day number, 1x8760 columns containing the month numbers, 1 x 8760 columns containing the years, and an array containing 1 x 8760 columns of average electrical energy usage data. The dates were split because the MatLab numeric arrays cannot contain text and therefore could not detect dates given in the format (DD/MM/YYYY).
- The time-series data is simply machine loading data that was obtained from the meter. This is the step in which the training function was chosen and the Levenberg-Marquardt learning algorithm was also selected because typically requires more memory but less time. Training automatically stops when generalization stops improving, as indicated by an increase in the mean square error of the validation samples. The learning Levenberg-Marquardt algorithm -works as a training algorithm with the capabilities of the pruning methodologies. Pruning is a process of examining a solution network, determining which units are not necessary to the solution, and removing those units [22]. The algorithm consists of finding the update by:

$$\Delta x = - \left[ J^T(x)J(x) + \mu I \right]^{-1} J^T(x) e(x) \quad (3)$$

where  $J(x)$  is the Jacobian matrix,  $\mu$  is a parameter conveniently modified during the algorithm iterations and  $e(x)$  is the error vector. When  $\mu$  is very small or null the Levenberg-Marquardt algorithm becomes

Loading [MathJax]/jax/output/CommonHTML/jax.js

convergence, however for higher  $\mu$  values, when the first term

within square brackets of Eq. (3) is negligible with respect to the second term within square brackets, the algorithm becomes steepest in descent [23].

## 2.2 Creation of a NARX network

The Nonlinear Autoregressive Network with External input (NARX) was created. This was a time-series network that predicted future data values using past data values.

Where the output of a data set was most certainly correlated to history of the input parameter, delayed variables were used. Instead of feeding it all at once, data was fed as a time series that is, it spread across the time dimension by augmenting the input with time-delayed copies of previous inputs. In this model, there were six input neurons which are: day of the week, hour, historical load, year, and previous 24-hour load. The output neuron was just the forecasted load. The layout of the neural network is shown in Fig. 4. The hidden layer had nineteen neurons.

### 2.2.1 Preparation of data for training

- The function PREPARETS was used to prepare the time series data for a particular network. It automatically shifted input and target time series as many steps as were needed to fill the initial input layer delay states. This meant that inputs were truncated by the number of delay states given.
- If the network had open-loop feedback, then it copied feedback targets into the inputs as needed to define the open-loop inputs. Using PREPARETES allowed one to keep the original time series data unchanged, while easily customizing it for the networks with differing numbers of delays with an open-loop or closed-loop feedback modes.

### 2.2.2 Selection of data division Function

- Setup Division of data for training, validation, and Testing. The Function DIVIDERAND was used to randomly assign target values to train, validation, and test sets during training with data dedicated for training being 70%, that dedicated for testing being 15% and that dedicated for validation being 15%.

## 2.3 Choosing of performance and plot functions

- The performance function chosen was the mean squared error.

$$J(w) = \frac{1}{n} \sum_{i=1}^n \left( y^{(i)} - f(x^{(i)}; W) \right)^2 \quad (4)$$

Where  $y^{(i)}$  is the actual and  $f(x^{(i)}; W)$  is the predicted outcome.  $W$  is the general collection of weights. The mean squared is only concerned with the average magnitude of error irrespective of their direction

Loading [MathJax]/jax/output/CommonHTML/jax.js

## 2.4 Training, Testing, and recalculation.

This is the step in which the network was trained, tested, and recalculated (validation), for performance measurement of the network. This process began with a network that had eight neurons in its hidden layer, and repeated, increasing the number of neurons up to twenty-three. The LM algorithm with nineteen neurons in the hidden layer for the network produced the best results, and it was used for generating the outputs. The relation between inputs and outputs was derived from the data itself, through a process of training that comprised of adjustments of weights linked to the connections, using learning algorithms. Figure 5 below is a snapshot showing the training process using the nntool tool.

During the learning process, a neural network builds 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 implies an optimization process. This error minimization process was repeated until an acceptable criterion for convergence was reached. To avoid overtraining and overfitting the early stopping method was used to stop the training process [24]. This was done because when the data was overfitted, the neural network stopped being intelligent instead it crammed.

The ANN performance was evaluated based on the shape of the load profile distribution and using the mean absolute per cent error (MAPE). The MAPE criterion is given by:-

$$MAPE = \frac{100}{N} \sum_{j=1}^N \frac{|p_j - a_j|}{\bar{p}} \quad (5)$$

Where  $p_j$  and  $a_j$  are the forecasted load and the actual load respectively at an hour  $j$ ,  $\bar{p}$  is the average price of the forecasting period and  $N$  is the number of forecasted hours.

## 2.5 Plotting of results.

The training results were then plotted against the target and the day ahead prediction against actual data was also plotted. After these steps were run down, the analysis of results showed that the prediction was not very far from the actual load on the same day. This showed that the NARX neural network is efficient and could be used to model accurate load forecasting.

## 3. Results And Discussion

This section reports the findings of the study based on the methodology that was applied to gather information. They are presented in figures of plots which are then analyzed and discussed.

### 3.1 Evaluation of training, testing, and validation

The 3 processes, training, testing, and validation were evaluated at once since the neural network plots the respective same functions same time, histogram error bins plots, regression plots, performance plots are all used to evaluate the process. In this section, only the regression plot and performance plot will be analyzed.

Figure 6 is used to validate network performance. The regression plots display the network outputs with respect to targets for training, validation, and test sets. For a perfect fit, the data should fall along the 45-degree line, where the network outputs are equal to the targets giving one(1) for the value of R. From Fig. 6 one can notice the closeness of the line of best fit to the line  $Y = T$ . For this problem, the fit is reasonably good for all data sets, with R values in training equal to 0.94614, in testing equal to 0.94358, and lastly 0.94532 in validation. The R-value for all of the three combined is equal to 0.94557. The initial R-value for all three combined was 0.91426. However, after training the network several times the R-value improved to 0.94557. Retraining will change the initial weights and biases of the network and may produce an improved network.

Performance is measured in terms of mean squared error and is shown on a log scale. It can be noted from Fig. 7 that the error decreased as the network was being trained. This implies that the accuracy of the network increased as the network was being trained (that is increase in the iteration number). Performance is shown for each of the training, validation, and test sets. This figure shows that training, testing, and validation errors all decreased until epoch thirty-five (35). At such an iteration, overfitting can be a problem, because neither testing nor validation error increased before iteration thirty-five(35). It is also advisable to stop training if the gradient remains the same for a while. In this case, the gradient did not change for almost an hour and thus training was stopped.

## 3.2 Prediction Performance

Figure 8 shows the plot of the predicted load profile against the actual load for the year. It can be noted from the training that the first few hours show that the forecasted load was way too low than the actual load on the ground. This shows that the network was inaccurate in predicting during the first few hours. The Zoomed portion for the first 600 hours is shown in Fig. 9.

### 3.2.1 Zoomed in portions of Fig. 8

Mean Absolute Percent Error (MAPE) for historical loads was found to be:-

$$MAPE = \frac{100}{N} \sum_{j=1}^N \frac{|p_j - a_j|}{\bar{p}} \quad (6)$$

$$MAPE = : 4.76\%$$

The value of MAPE obtained for the whole load profile from 16-01-2018 to 15-01-2019 is slightly higher

Loading [MathJax]/jax/output/CommonHTML/jax.js

than what has been in other models that have been found online. [25] found a MAPE of 1.86% for the whole load profile which is more accurate compared with the value that was obtained from this research.

### 3.3 Performance of the neural network for the day ahead forecast (16-01-2019).

From the graph, one can note that for the first few hours of the day, the load was accurately forecasted as seen from the third hour that there was a very little deviation of the forecasted demand. However, there was a big deviation between the forecasted load and the actual load as the day progressed, this may have been due to different conditions that may have contributed to low demand, things like weather and GDP. The mean absolute per cent error (MAPE) for the forecasted load for the day ahead was found to be:-

$$MAPE = \frac{100}{N} \sum_{j=1}^N \frac{|p_j - a_j|}{\bar{p}} \quad (7)$$
$$MAPE = 6.71\%$$

The value of MAPE obtained for the day (16-01-2019) is slightly higher than what has been in other models that have been found online. [25] found a MAPE of 1.84% for the day ahead forecast which is more accurate compared with the value that was obtained from this research. This may be because they used other data sets like electricity prize and so on which may have contributed to the accurateness of their work.

## 4. Conclusion

A good model has been designed that can-do short-term Load forecasting and can help power utilities to work efficiently and provide demanded loads with less discrepancy between demand and supply. Forecast hourly and daily energy consumption can be useful to determine the required size of a storage energy system, delay and postpone energy consumption, and can be used at the renewable energy system early design stage. It can also help the demand-side management (DSM), such as electricity suppliers, to forecast the likely future development of electricity demand in the entire sector of the community. The Levenberg-Marquardt algorithm typically requires more memory but less time. Training automatically stops when generalization stops improving, as indicated by an increase in the mean square error of the validation samples. Outliers in the load data-set affect the general performance of the network, these outliers are considered when formulating the network error. The designed system was able to predict the load with minimum error, meaning the model is adequate and can be implemented.

For more accurate load forecasting results. It is recommended to use more than one input parameter as a predictor. For instance, weather data could have made the predictions more accurate.

A larger data-set is also recommended to be used to make the results more sensible, unlike in this research where data from one feeder at a substation was used. This will also enable the network to distinguish between good and bad data input for network generalization purposes. In future, data logging can be improved by integrating the internet of things (IoT) technology and an online database connectivity server for more reliable information, moving towards online backup systems.

## Declarations

## Conflict of interest

The authors declare no conflict of interest.

## Funding

The authors did not receive support in funds or grants from any organization for the submitted work.

## Availability of data and material

Data used in this research is available and was obtained from ZETDC.

## Code Availability

The code of the artificial neural network that was designed is also available.

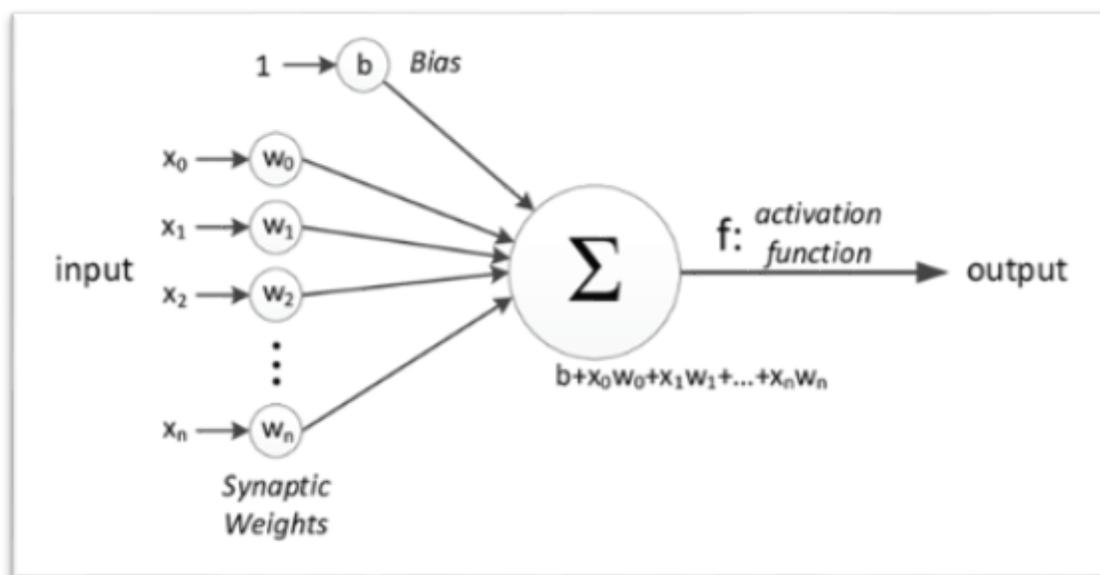
## References

1. Bansal S, Lodhi RS, Nema P, "State of Art on Short term Load Forecasting Using Artificial Neural Network," IOSR Journal of Electrical and Engineering E (IOSR-JEEE), vol. 13, no. 3, pp. 80–85 (2018) DOI: 10.9790/1676-1303028085
2. Energypedia "ZimbabweE Situation," 2020. [Online]. Available: [https://energypedia.info/wiki/Zimbabwe\\_Energy\\_Situation](https://energypedia.info/wiki/Zimbabwe_Energy_Situation). [Accessed 24 March 2020]
3. ZPC, "Zimbabwe Power Company," 21 January 2020. [Online]. Available: <http://www.zpc.co.zw>. [Accessed 21 January 2020]
4. Sheikh K, Samsher, Unde M (2012) Short-term load forecasting using ANN technique. International Journal of Engineering Sciences Emerging Technologies 10(1):1–5. doi:10.7323/ijeset/v1\_i2\_12\_4
5. Lin J, Magnago FH (2017) Electricity Markets, 1st edn. John Wiley & Sons, South Africa

6. Kuster C, Rezgui, Yacine and Mourshed and Monjur, "Electrical load forecasting models: A critical systematic review," *Sustainable cities and Society*, vol. 35, no. 10, pp. 2–4, 2017
7. Hesham A-A, Elham E, Sameem AK (April 2018) "A review of smart meter load forecasting techniques. *Scale and Horizon*," London
8. Srivastava AK, Pandey AS, Singh D, "Short-term load forecasting Methods: A review," *International conference on Emerging trends in Electronics & sustainable Energy systems*, 2016
9. Hsu C-C, Chen, Chia-Yon (2003) "Regional load forecasting in Taiwan-applications of neural networks". *Energy conversion Management* 44(12):1941–1949. doi:10.1016/S0196-8904(02)00225-X
10. Soni M, "Quality assessment of geographical based load forecast technique at Eskom Distribution: Forecast error," in 8th Southern Africa regional conference, South Africa, 2017
11. Maronna RA, Martin RD (2006) *Robust Statistics: Theory and Methods*, 2nd edn. Wiley, Chichester ISBN: 0470010924 9780470010921\$4
12. Kuo P-H, Huang C-J, "A high Precision Artificial Neural Networks Model for Short-term energy load Forecasting," *MDPI Energies*, vol. 11, no. 213, pp. 1–8, 2018
13. Park DC, El-Sharkawi MA, Marks RJ, Atlas, Damborg MJ, "Electric load forecasting using artificial neural network," *IEEE transactions on Power Systems*, vol. 6, no. 2, pp. 442–449 (1991) **DOI: 10.1109/59.76685**
14. Adepoju G, Ogunjuyigbe M, Ayodeji, Alawode, Ayodegi and Alawode and Kehinde and Tech B, "Application of Neural Network to Load Forecasting in Nigerian Electrical Power System," *The Pacific Journal of Science and Technology*, pp. 68–72, 2007
15. Nagesh SC, "Towards data science," (2019) [Online]. Available: <https://towardsdatascience.com/introduction-to-artificial-neural-networks-ann-1aea15775ef9>. [Accessed 7 February 2020]
16. Karlik B, Olgac, Vehbi A (2011) "{Performance analysis of various activation functions in generalized MLP architectures of neural networks". *International Journal of Artificial Intelligence Expert Systems (IJAE)* 1(4):111–112
17. Sharma S, "towards data science," (2017) [Online]. Available: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>. [Accessed 15 February 2020]
18. Amini A, "MIT 6.S191," 27 January 2019. [Online]. Available: [http://introtodeeplearning.com/slides/6S191\\_MIT\\_DeepLearning\\_L1.pdf](http://introtodeeplearning.com/slides/6S191_MIT_DeepLearning_L1.pdf). [Accessed 26 February 2020]
19. Wythoff BJ, "Backpropagation neural networks: A tutorial," *Chemometrics and intelligent laboratory Systems*, vol. 18, no. 2, pp. 115–155 (1993) <https://doi.org/10.2307/1270603>
20. Sisense "SISENSE," (2020) [Online]. Available: <https://www.sisense.com/glossary/data-cleaning/>. [Accessed 8 June 2020]

21. Hawkins DM, in Identification of outliers, Cape Town, Springer, 1980, pp. 27–28. ISBN 978-94-015-3994-4
22. Dow RJ, Sietsma J, "Neural Net Pruning: Why and How. In Proceedings of IEEE int Conf. Neural Networks.," vol. 1, pp. 325–333, 1988
23. Saini LM, Soni MK, "Artificial neural network based peak load forecasting using Levenberg-Marquardt and quasi-new methods IEE Proceedings General Transmission Distribution.," vol. 5, no. 149 (2002) DOI: 10.1049/ip-gtd:20020462
24. Sarle WS; "Stopped training and other remedies for overfitting. Proceedings of the 27th symposium on the interface of computing science and statistics." Malaga (2010) DOI: 10.1007/978-3-642-00267-0\_1
25. Deoras A, "Matlab central file exchange," 1 September 2016. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/28684-electricity-load-and-price-forecasting-webinar-case-study>. [Accessed 20 July 2020]

## Figures



**Figure 1**

Single layer perceptron showing inputs, bias and synaptic weights (Nagesh, 2019)

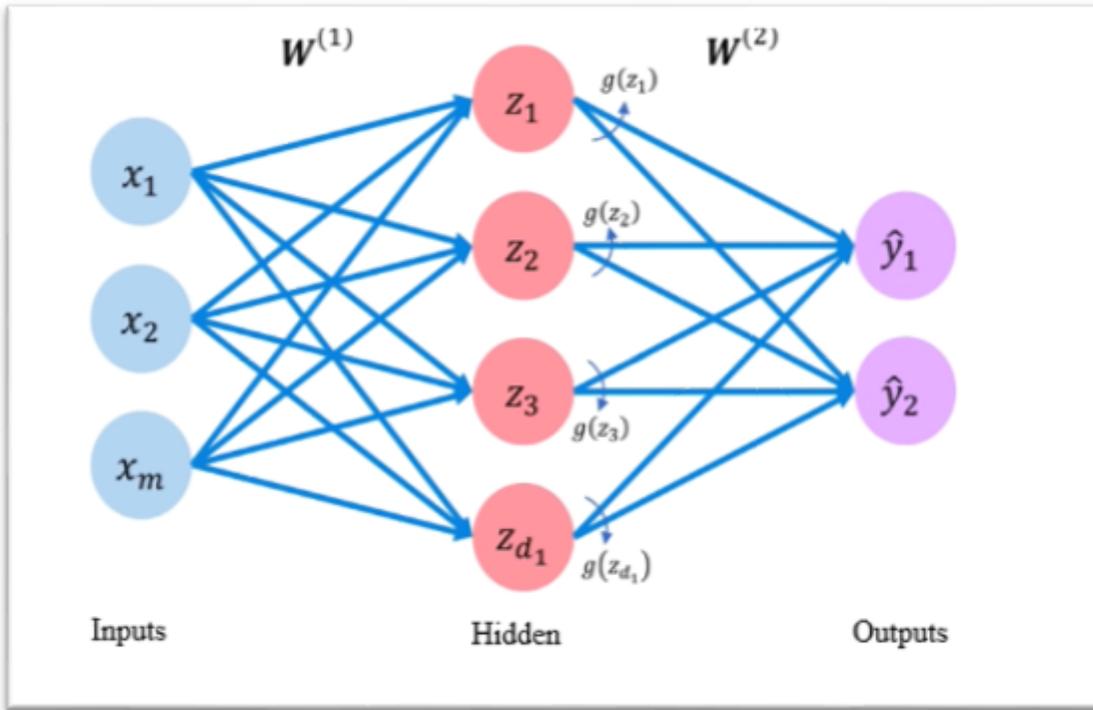


Figure 2

Multi-layer perceptron with multiple outputs showing the inputs layer, the hidden layer and output layer (Amini, 2019)

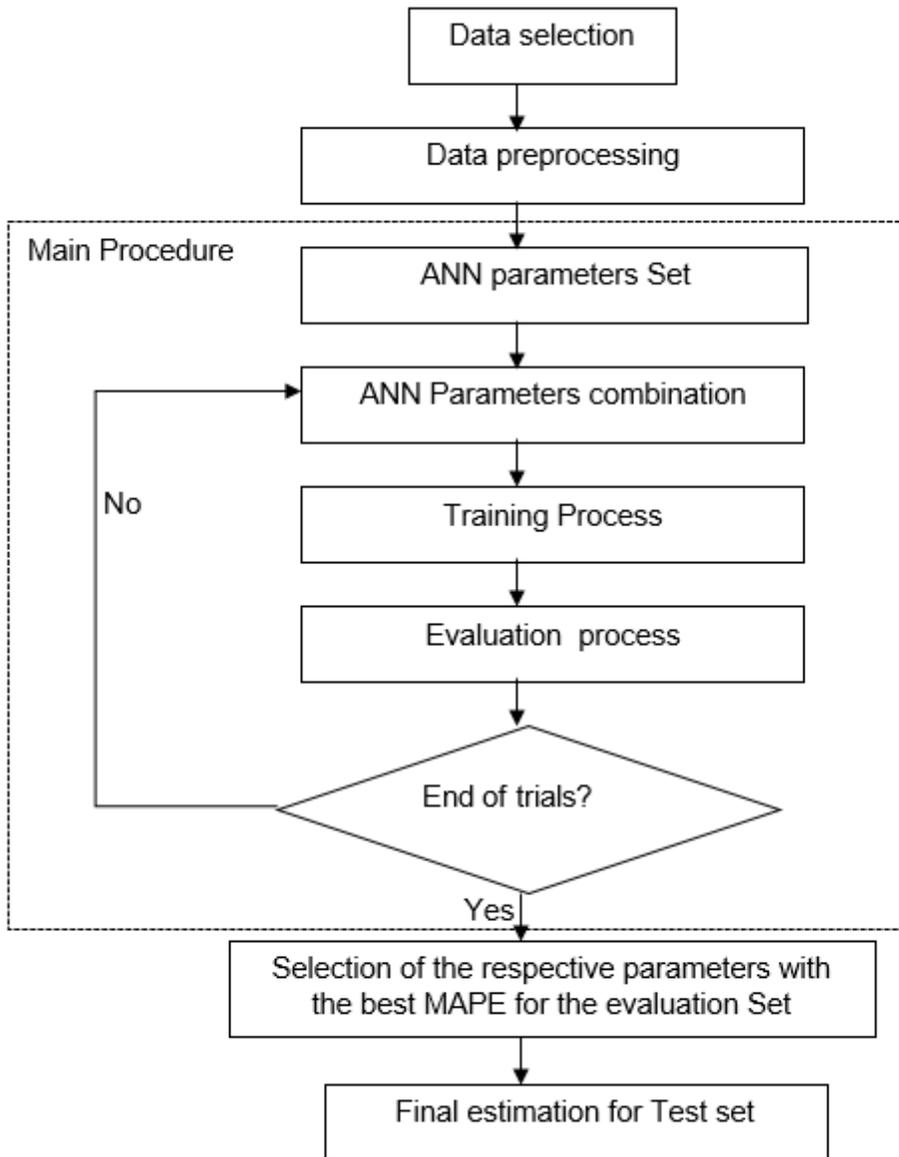


Figure 3

Methodology flow chart outlining the steps that were taken while designing the Artificial neural network

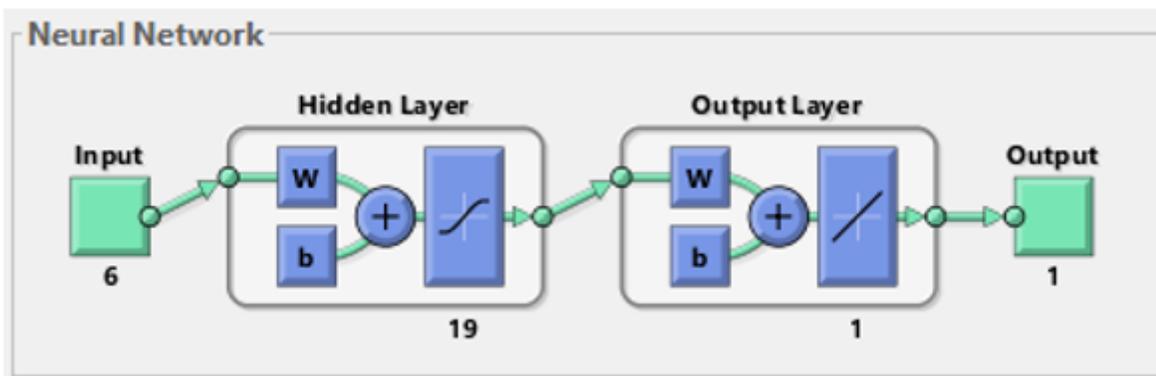


Figure 4

Loading [MathJax]/jax/output/CommonHTML/jax.js

Structure of the neural network showing 6 input neurons, 19 neurons in the hidden layer and one output neuron

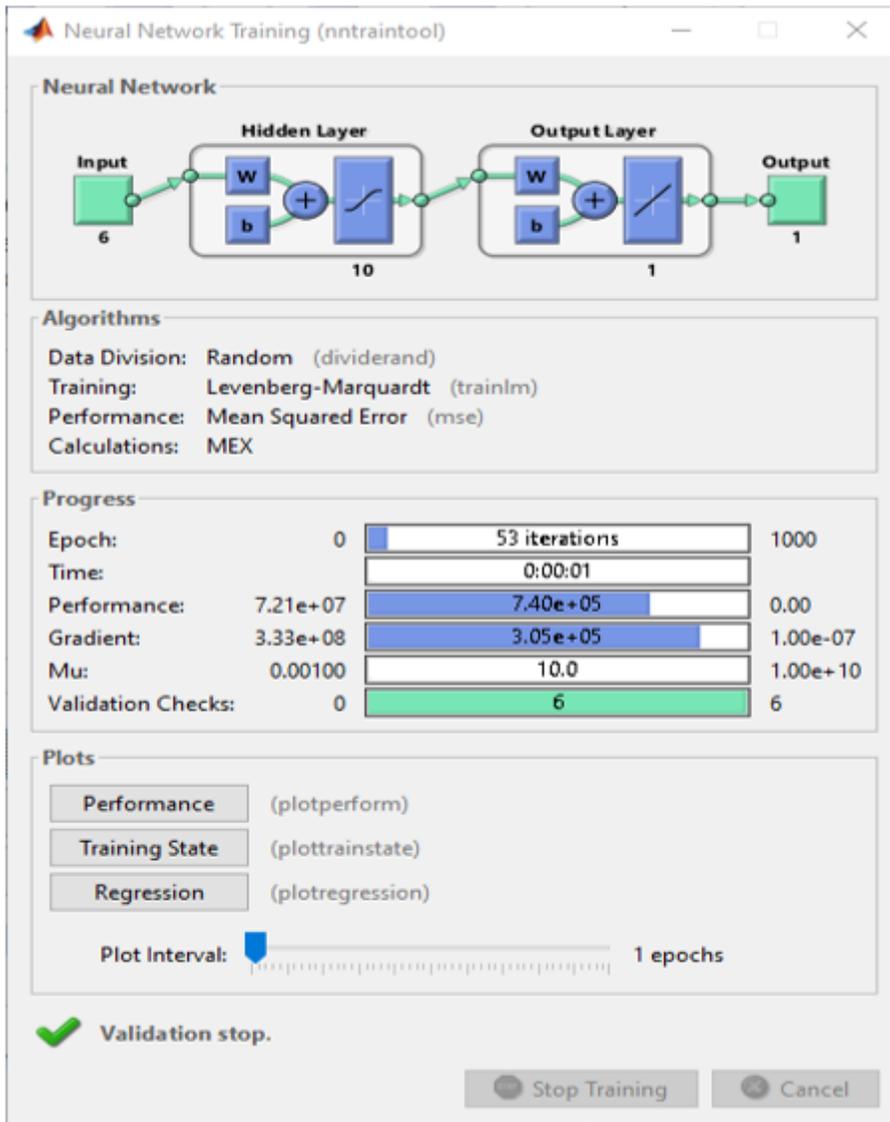


Figure 5

network training with 10 hidden neurons

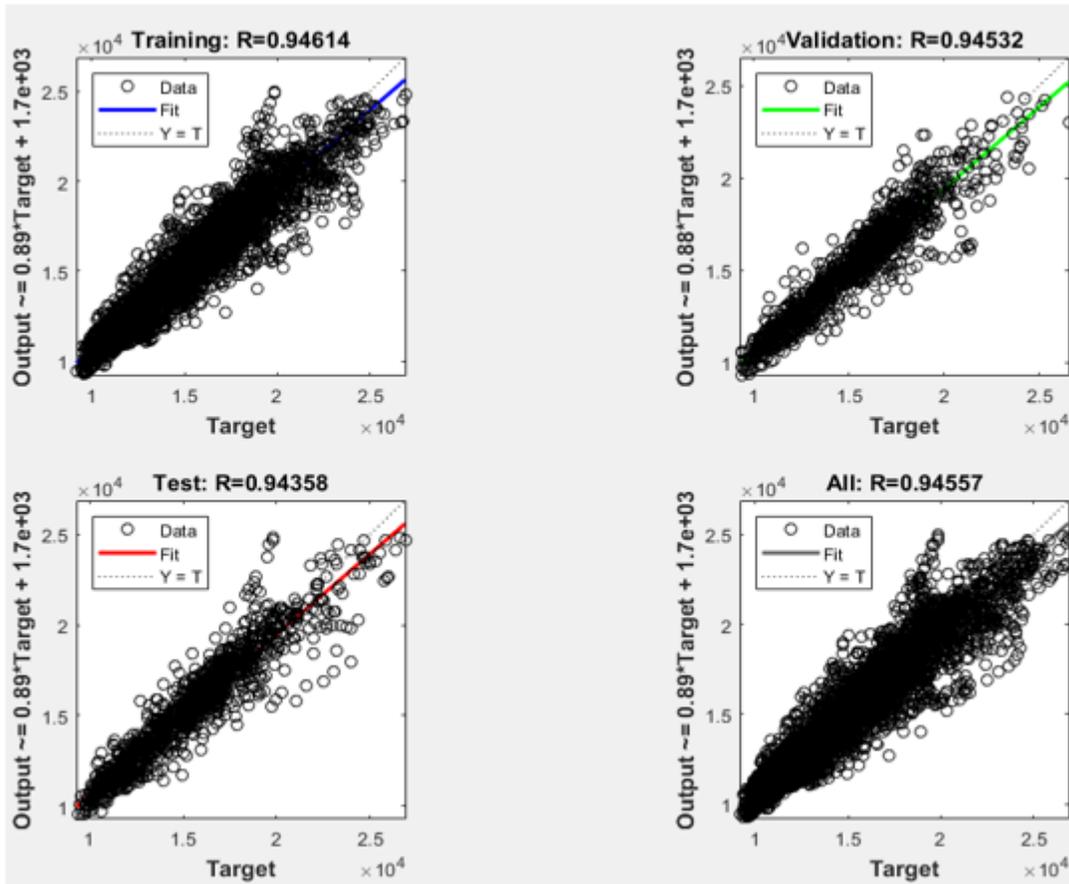
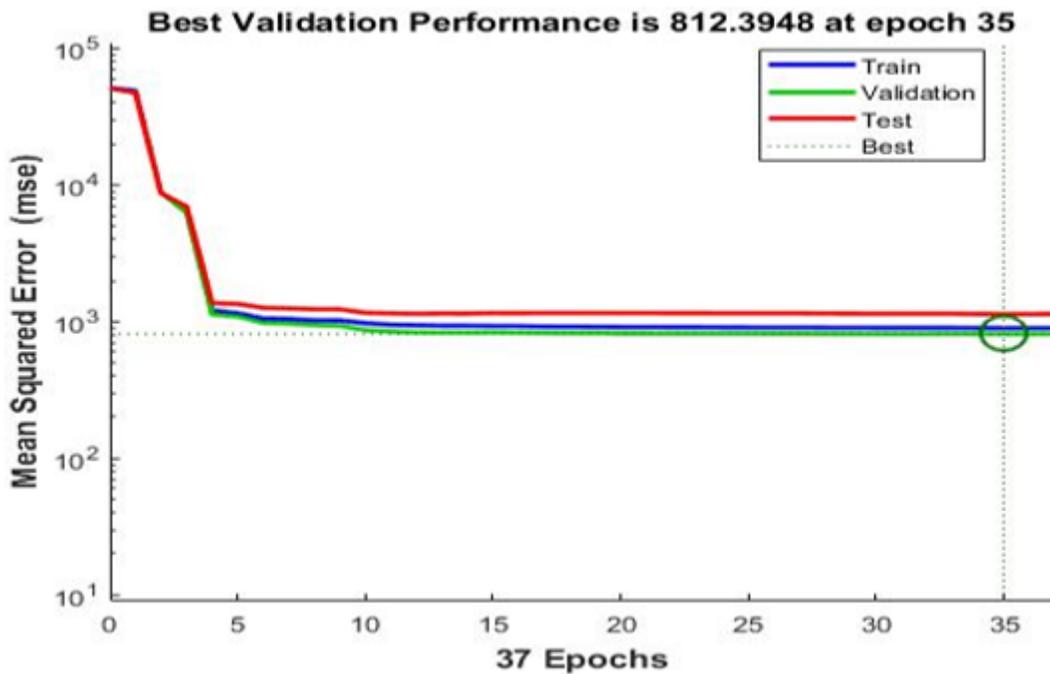


Figure 6

Regression plots for Training, validation, testing, and the overall plot



Network performance plot showing the train, validation and test performance curves which decreased until epoch number thirty-five

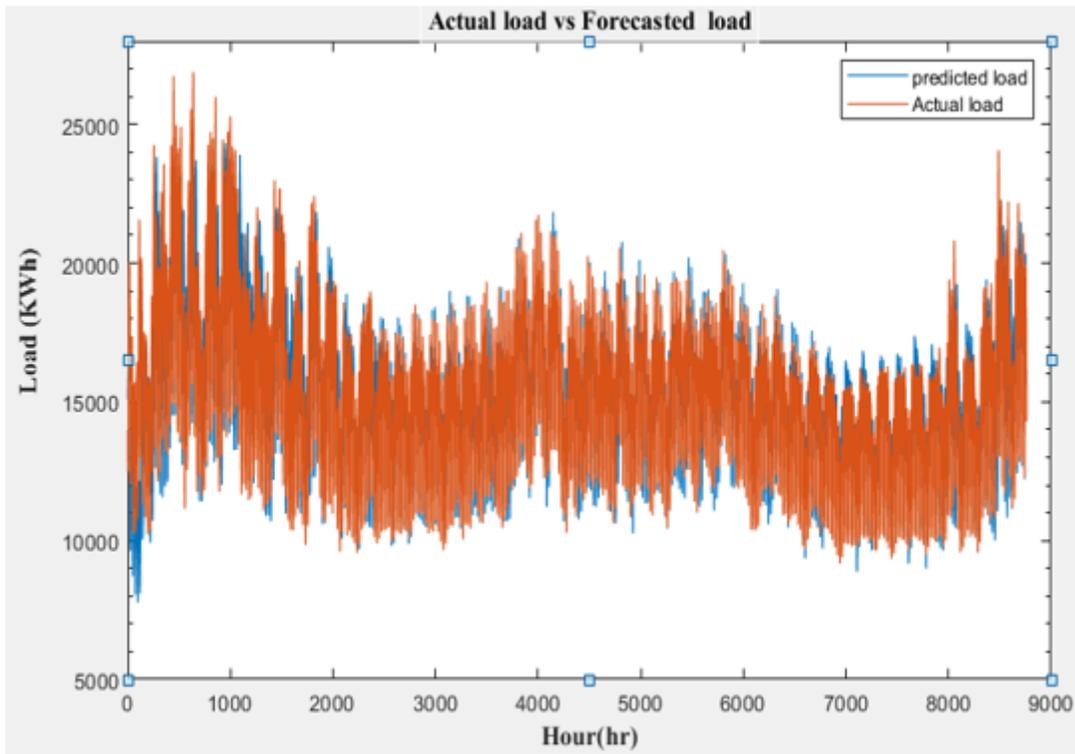


Figure 8

Actual load vs Forecasted load for 1 year

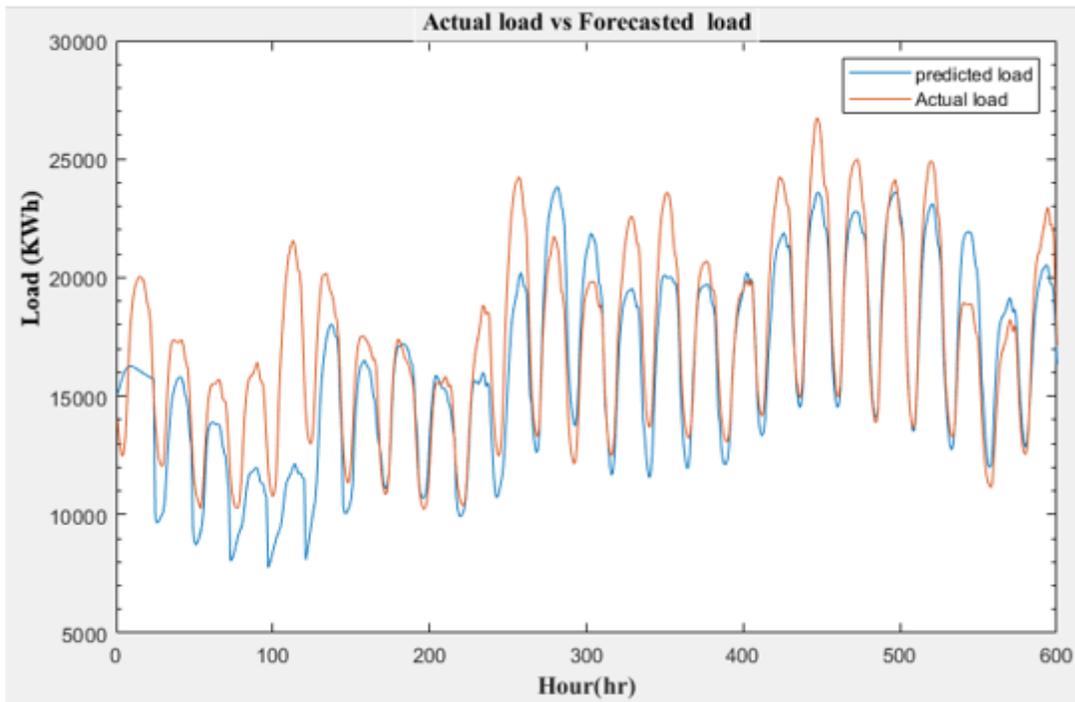


Figure 9

Loading [MathJax]/jax/output/CommonHTML/jax.js

Actual load vs forecasted load for the first 600 hours

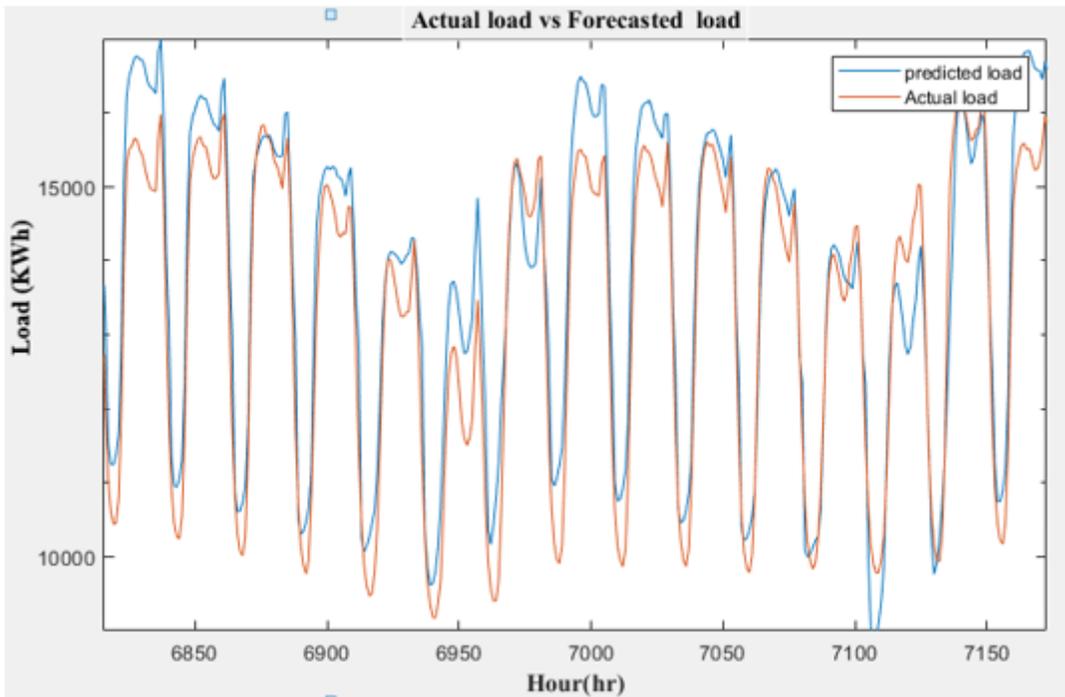


Figure 10

forecasted load for the 6800th hour and 7150th hour

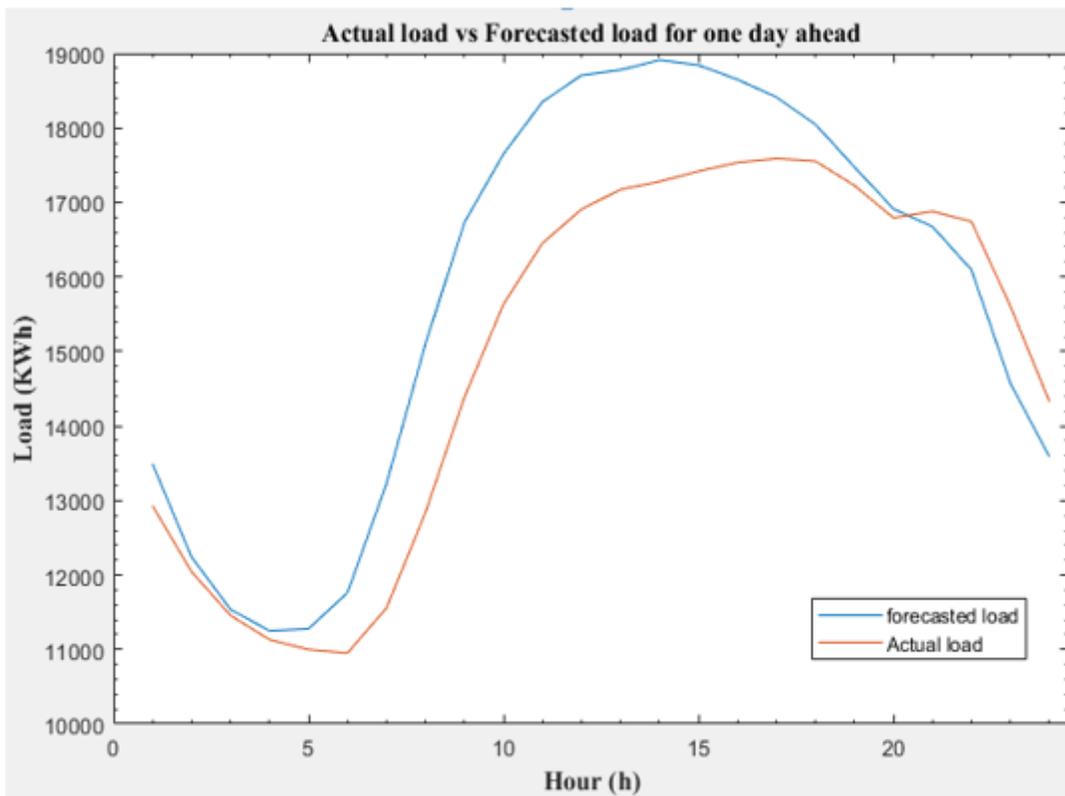


Figure 11

Loading [MathJax]/jax/output/CommonHTML/jax.js

plot of actual load vs forecasted load for 16-01-2019.