Predicting Electricity Load For Light Energy Consumption In House Using Machine Learning Algorithm

M.Jona Fark[1], B.S.E.Zoraida[2]
[1]Dept of Computer Science
[2]Assistant Professor, Dept of Computer Science
[1,2]School of Computer Science, Engineering and Applications, Bharathidasan University, Tiruchirappalli Tamilnadu, India

Abstract- *Predicting energy consumption is an area has become an important field of research. The purpose of this research work is to find the relation between the usage of light energy consumption with outside temperature and humidity. In this work, time series modeling data of electric light energy consumption for a single house is considered from January 2016 to May 2016. Artificial neural network has various time series models for predicting time series data such as NAR and NARX. In this work, the error rate and accuracy value of NAR and NARX time series models are compared. NARX model provides better results when compared to NAR model. Artificial neural network has three different training algorithms like Levenberg-Marquardt, Bayesian Regularization and scaled Conjugate Gradient algorithms. These algorithms have different levels of accuracy value and error rate for the observed data and predicted data. Bayesian Regularization training algorithm provides the best accuracy and error rate for light energy consumption.*

Keywords- Prediction, Light energy consumption, time series model, performance matrices, seasonal energy prediction, training algorithm.

I. INTRODUCTION

Energy is a necessary part of our lives and all things associated with electricity [1, 2]. Good lighting creates a visual that enable people to see, to move one place to another place safely during night time. The light energy considered maybe daylight, electric light or both. Predicting future trends used it many applications, time series prediction data were used to identify the light energy consumption and occupant behaviour in the past. In this paper, prediction is made on light energy consumption with respective to weather condition. The purpose of those predicted to understand the time series modeling with training algorithms, and neural network. Section 2 shows the related work of the research, followed by section 3 describes the house description and experimental setup, and the section 4 shows the research methodology and finally Section 5 shows results and conclusion.

II. LITERATURE REVIEW

The current various researches have used the method of predicting with time series data such as the energy consumption. Kalogirou [3] proposed a back propagation neural network for heating load prediction in buildings. These building energy consumption data of 225 buildings were different in size. Olofsson [4] approached a method to predict the energy on a yearly based for a single house in Sweden. Irisarri [5] approached a method of energy prediction based on summer and winter seasons. Dong et al. [7] developed an SVM-based energy consumption prediction model. They predicted monthly electricity consumption based on outdoor bulb temperature, relative humidity, and global solar radiation. D.C. Park et al [8] proposed an electric load forecasting using ANN model between load and temperature. They compared the average of errors for 24 hours

III. HOUSE DESCRIPTION AND EXPERIMENTAL SETUP

The analysis of this energy data is done in one house is located in Stambruges - City of Mons in Belgium. The electrical light energy metering was done with the M-Bus energy counter. The energy information collected and reported by e-mail every 12hours. The light energy consumption (watts) in a house has been studied by every 10 minutes. The 10 min interval was to capture quick changes in energy consumption. The use of lights in the house depends upon the outside building temperature (Celsius) and humidity (%). Humidity and temperature are monitored by a wireless sensor network.

The combined data set is 137days (4.5months). Outside of the house temperature and humidity is merged by date and time. Complete dataset energy consumed from 11, January until 27, May is split into winter (81days) and spring (56days) season-wise. Four occupants in the house, two adults and two teenagers. One of the adult works regularly in the home office.

IV. METHODOLOGY

4.1 ANN models

The Artificial Neural Network model involved five steps. There are 1) collecting data, 2) preprocessing data, 3) building network, 4) train network, 5) test performance as shown in Fig: 4.1

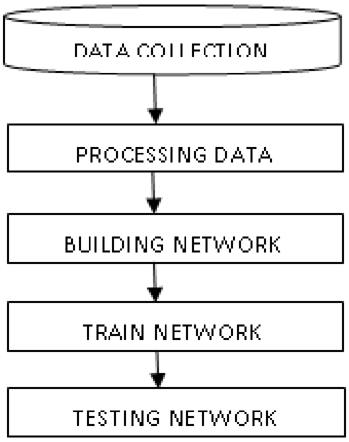

Fig: 4.1 - Basic Flow for ANN model

Artificial Neural Network (nnstart)

Artificial neural network has provided an exciting alternative method for solving a variety of problems in different fields like science and engineering. The Neural network can be divided into Biological model and artificial model. Artificial systems knowledgeable of difficult calculation comparable to the human brain. It is composed a large number of highly interconnected processing elements called neurons. An ANN is Specific application such as pattern recognition, times series modeling app, clustering and fitting.

4.2 Neural Time Series(ntstool)

Prediction is a kind of dynamic filtering, in which past values of one or more time series are used to predict the future values. There are many applications for prediction.

Syntax for NTS, ntstool - opens the neural network time series tool. ntstool ('close') - closes the tool

NTS tool have some time series model to find the best predictor

NARX model is the best prediction model for time series modeling. Nonlinear Autoregressive with External Input(NARX) - define inputs and targets as x(t) and y(t) and the time series format as Matrix row. NARX network is predict one time series given past values of same time series, the feedback input and rest of called exter time series.

Syntax for NARX, narxnet (inputDelays. feedbackDelays,hiddenSizes,trainFcn)

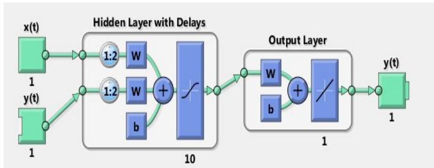

Fig: 4.2.1 NARX network closed loop

The network Created and trained in open loop form as shown in Fig:4.2.1. Open loop is more efficient than closed loop training. Open loop allows to supply the network with correct past outputs as train it to produce the correct current outputs. After training, the network may be converted to closed loop form or any other form.

Validation and testing - Target timesteps 9867 data are randomly divide up. Three kinds of Target Timesteps:

1. Training: These are presented to the network during training, and the network is adjusted according to its error
2. Validation: These are used to measure network generalization, and to halt training when generalization stops removing.
3. Testing: This testing have no effect on training and so provide an independent measure of network performance during and after training.

Network Architecture - This step is involved in change the number of neurons or delays if the network does not perform well after training.

NN training Tool Plots

1. Plot Error Histogram of error values e. it takes number of error and names and plots.

The bin defines to use histogram plot. The default value is 20bin as shown in Fig:4.2.2.

Syntax for Plot error histogram, ploterrhist(e) ploterrhist(e1,’name1’,e2,’name2’,….) ploterrhist(..,’bins’,bins)

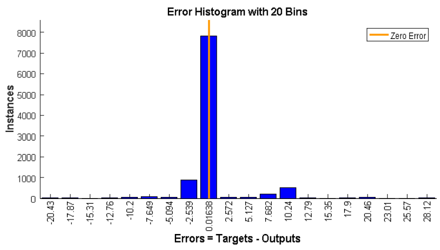

Fig:4.2.2 plot error histogram for BR

2. Error Auto correlation is an error time series and plots the autocorrelation of error varying Lags. It use the property name or value define output error auto correlation plotted. The default value is 1 as shown in Fig:4.2.3

Syntax for Auto coorelation, ploterrcorr(error)

Ploterrcorr(errors,’outputIndex’,outIdx)

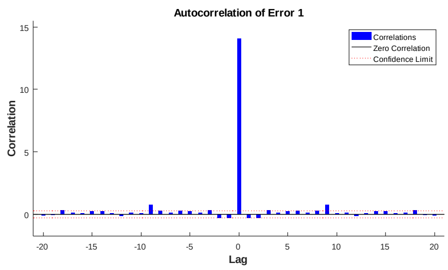

Fig:4.2.3 plot error auto correlation for BR

3. Time-Series Response is a dynamic network. It takes the target times series is an output(y) and plot the same axis for showing the errors. Plot the response with different colors indicate the target. The default value is 1 as shown in Fig:4.2.4

Syntax for Time-Series response, plotresponse(t,y)
plotresponse(t1,’name’,t2,’name2’,….)
ploterrhist(...,’outputIndex’,outputIndex)

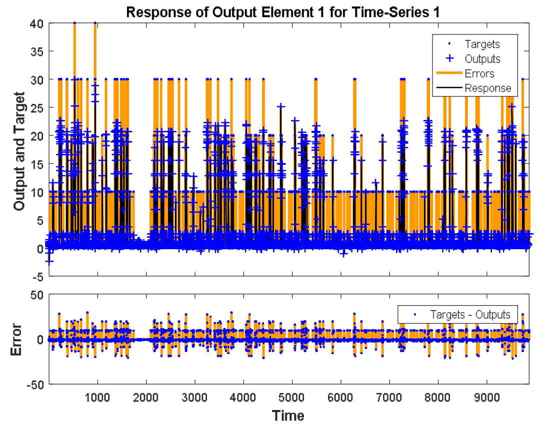

Fig:4.2.4 plot Time-series response for BR

4. Input-error cross-correlation, it takes an input series(x) and error time series e. it also varying the Lag. The default value is 1 as shown in Fig:4.2.5.

Syntax for Input-error cross-correlation, plotinerrcorr(e)

Plotinerrcorr(…,’inputIndex’,inputIndex)

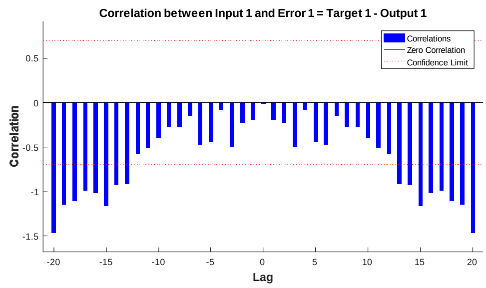

Fig:4.2.5 plot input-error cross-correlation for BR

4.3 Training networks

4.3.1 Levenberg Marquardth(trainlm)

The LM, also known as the damped least-squares method used to solve non-linear least square problem. This algorithm requires more memory but less time. Training automatically stops, as indicated by an increase in the mean square error of the validation. Train network multiple times will generate different results.

Syntax for LM, net.trainFcn = 'trainlm'
[net,tr] = train(net,...)

4.3.2 Bayesian Regularization (trainbr)

The trainbr is a network training function that updates the weight and bias values according to Levenberg-

Marquardt optimization. BR minimizes the combination squared errors and weights, then determines the correct combination, to produce a network that generates well. This algorithm requires more time, but result in good. The process is called Bayesian regularization.

Syntax for BR, net.trainFcn = 'trainbr'
[net,tr] = train(net,...)

4.3.3 Scaled conjugate gradient(trainscg)

The trainscg is a network training function that updates weight and bias values according to the scaled conjugate gradient method. This algorithm requires less memory.

Syntax for SCG, net.trainFcn = 'trainscg'
[net,tr] = train(net,...)

4.4 evaluate network

Test network on more data and then decide network performance is good enough. If a try did not generate the good results try training again and get the best result

4.5 Performance Matrices

4.5.1 Mean Squared Error (MSE)

Mean squared Error is an estimator to measure the average of the squares of the errors – that is, the average squared difference between the outputs and targets. The algorithm which gives less value is the best algorithm. In this implementation Bayesian Regularization algorithm gives less error as shown in Fig:4.5.1

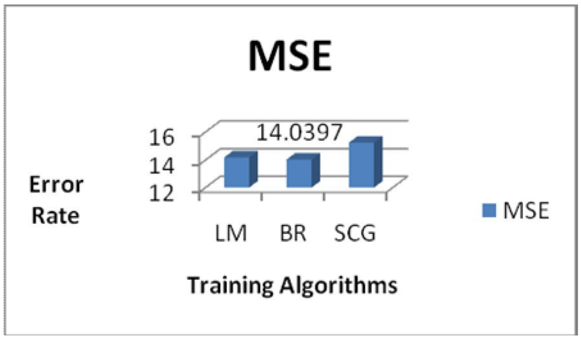

Fig: 4.5.1 Mean Square Error for Multiple Algorithm

4.5.2 Regression (R):

Regression Values measure the correlation between outputs and target. If the R value is 1 it is inferred to have a close relationship, 0 a random relationship.

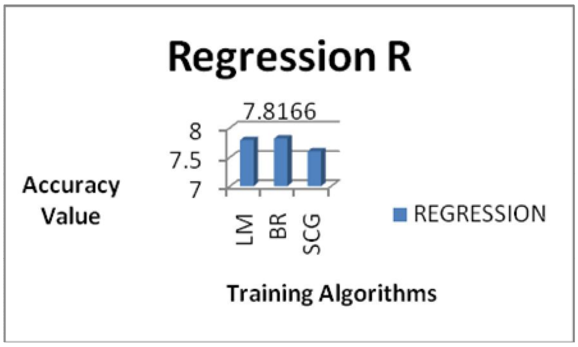

Fig: 4.5.2 Regression R accuracy for Multiple Algorithm

Fig:4.5.2 shows the Regression accuracy for the network performance. In Bayesian Regularization gives high accuracy rate 7.8166.

Table 1: MSE and REGRESSION R values for training algorithms

Training Algorithm	MSE	Regression R
Levenberg Marquardth	14.18727	7.79135
Bayseian Regularaization	14.0397	7.8166
Scaled Conjugate Gradient	15.24486	7.60052

4.5.3 Mean Absolute Percentage Error (MAPE):

Mean Absolute Percentage Error is a simple average of percentage error. It is also referred to as MAPD; This error is used to measure the accuracy for constructing the fitted time series.

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

MAPE= mean((abs(V2-V1))./V1)
----- eqn (1)

V1= Observed value
V2= Predicted value

4.5.4 Root Mean Square Error (RMSE):

RMSE is the average magnitude of the error. It’s the square root of the average of square differences between prediction and actual observation.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - o_i)^2}{n}}$$

RMSE = sqrt(mean((V1-V2).^2)) ----- eqn (2)

V1= Observed value
V2= Predicted value

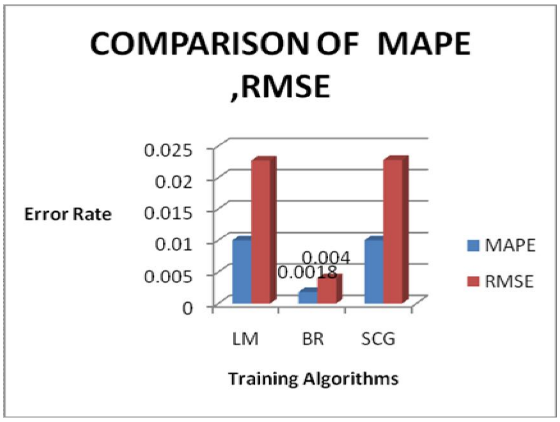

Fig: 4.5.3 MAPE and RMSE error rate for Multiple Algorithm

Fig:4.5.3 shows the MAPE and RMSE error rate for the network performance. In Bayesian Regularization gives low error rate for both mape=0.0018 and rmse=0.0004.

Table 2: Comparison for MAPE,RMSE values for training algorithms

Training Algorithm	MAPE	RMSE
LevenbergMarquardth	0.01	0.0227
Bayseian Regularaization	0.0018	0.004
Scaled Conjugate Gradient	0.01	0.0228

4.5.5 Result of Time Series Modeling

Bayesian Regularization – Observed Data Vs Predicted Data

Comparison of Light energy consumption between Observed data in the house and Predicted data as shown in Fig: 4.5.5.

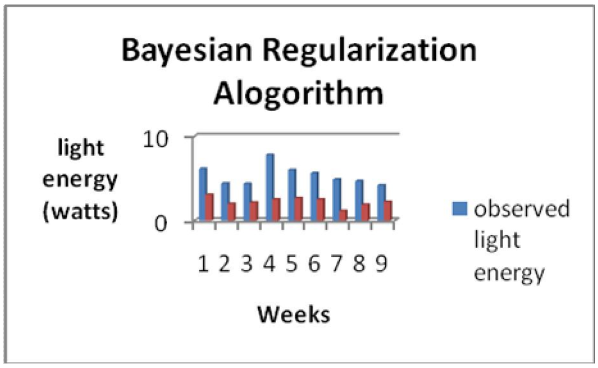

Fig: 4.5.5 Observed data Vs Predicted data for BR algorithm

V. CONCLUSIONS

In this research work, MATLAB tools are used to predict the light energy consumption of Stambruges, City of Mons – Belgium. In various time series prediction models, NARX is the best model to find the future (data) light energy consumption of the house. In training network algorithms, Bayesian Regularization is the best resulting network and it provides the predicted value with minimum error value. The error values are less when compared to other algorithms. In light energy prediction, BR is the best and good resulting training algorithm.

REFERENCES

[1] Fayaz, M.; Kim, D. Energy Consumption Optimization and user comfort management in residential buildings using a bat algorithm and fuzzy logic. Energies 2018, 11, 161.

[2] Selin, R. The Outlook for Energy: A View to 2040; ExxonMobil: Irving, TX, USA, 2013

[3] Kalogirou, S.; Neocleous, C.; Schizas, C. Building heating load estimation using artificial neural networks. In Proceedings of the Clima 2000 Conference, Brussels, Belgium, 30 August–2 September 1997.

[4] Olofsson, T.; Andersson, S.; Östin, R. A method for predicting the annual building heating demand based on limited performance data. Energy Build. 1998, 28, 101–108.

[5] Irisarri, G.; Widergren, S.; Yehsakul, P. On-line load forecasting for energy control centre application. IEEE Trans. Power App. Syst. 1982, 71–78.

[6] B Dong, C Cao, SE Lee. Applying support vector machines to predict building energy consumption in tropical region, Energy Build. 37 (2005) 545-553.

[7] D.C.Park,M.A.El-sharkawi, R.J.Marks II, L.E.Atlas and M.J.Damborg, “Electric Load Forecasting Using An Artificial Neural Network” , IEEE Transactions on Power Engineering, vol.6, pp.442-449 (1991).

BIOGRAPHIES

Ms. M.Jona Fark is the research scholar, pursuing M.Phil in school of computer science, engineering and application, Bharathidasan University, Tiruchirappalli, India. She received her M.Sc Computer Science Degree in Bharathidasan University, Trichy.

Dr. B.S.E. Zoraida presently she is an Assistant Professor in school of computer science, engineering and application, Bharathidasan University, Tiruchirappalli, India. Her area of research includes computer application in smart grid and molecular computing.